

manual

Delft-ID2D **SOBEK Overland Flow Module**

Version 2.0

483

November 2000



Ref. nr: 005869

wL | delft hydraulics

Delft-ID2D

SOBEK Overland Flow Module

Version 2.0

November 2000



wL | delft hydraulics

OPDRACHTGEVER: Rijkswaterstaat DWW




TITEL: User manual Delft-1D2D

SAMENVATTING:

In januari 1998 sloten RWS Dienst Weg- en Waterbouwkunde (DWW) en WL| Delft Hydraulics samenwerkingsovereenkomst DWW 1382. In het kader van deze overeenkomst en de vervolgovereenkomst (DWW 1613) hebben DWW en WL gezamenlijk een aantal onderzoeks- en ontwikkelingsprojecten uitgevoerd, gericht op de koppeling tussen Delft-FLS en SOBEK. Het uiteindelijke product van de samenwerking is de nieuwe 2D overland flow module binnen SOBEK. Deze module is gericht op studies met onderwerpen als rampenbestrijding (bij overstromingen), evacuatieplanning, schadeberekeningen en risicoanalyse. De 2D overland flow module vormt samen met de Channel Flow module van Sobek Rural 'Delft-1D2D'.

Het voorliggend rapport is de user manual van de 2D overland flow module. Het bevat een introductie in de basisprincipes van de 2D module, een beschrijving van de basisfunctionaliteiten en de nieuwe functionaliteiten binnen het SOBEK user interface, een lijst van 'Frequently Asked Questions' en een tutorial. Deze manual (versie 2) behoort bij de oplevering van versie 2.30.000.10 van Delft-1D2D.

REFERENTIES:

VER.	AUTEUR		DATUM	OPMERK.	REVIEW		GOEDKEURING	
1	Ir. E. Verschelling		29 augustus 2000	-	Ir. M. Laguzzi		Prof. ir. E. van Beek	
2	Ir. E. Verschelling		2 november 2000	-	Ir. N.N. Lorenz		6/a P. Scaen	
PROJECTNUMMER:			Q2755					
TREFWOORDEN:								
INHOUD:	TEKST	X	TABELLEN	X	FIGUREN	X	APPENDICES	X
STATUS:		<input type="checkbox"/> VOORLOPIG		<input type="checkbox"/> CONCEPT		<input checked="" type="checkbox"/> DEFINITIEF		

Contents

1	Introduction.....	1-1
1.1	General.....	1-1
1.2	Outline of this report.....	1-1
1.3	Definitions.....	1-2
1.4	Height or Depth?	1-2
2	Background principles	2-1
2.1	Basic Equations used for 2D computation.	2-1
2.2	Equation Solving Method	2-2
2.3	1D-2D Connection	2-2
2.4	2D-2D Connection	2-3
3	Description of every task block.....	3-1
3.1	Task block 1: 'settings'	3-1
3.1.1	General	3-1
3.1.2	Overland flow settings	3-2
3.2	Task block 2: 'schematisation', Edit model	3-8
3.2.1	General	3-8
3.2.2	Edit network mode.....	3-9
3.2.3	Edit model data mode	3-23
3.3	Task block 3: 'results in maps'	3-33
3.3.1	General	3-33
3.3.2	Viewing model and output data in Netter.....	3-33
3.3.3	Incremental output files	3-34
3.3.4	GIS/ MAP output files	3-34
3.3.5	Output for 1D dam break reach	3-39
4	Frequently Asked Questions.....	4-1
4.1	Modelling-related questions.....	4-1
4.1.1	When do I choose dikes in the 1D profiles?.....	4-1

4.1.2	How do I visualize the bottomlevels of the 2D grid together with the bottom level of the profiles?	4-2
4.1.3	How do I model culverts	4-3
4.1.4	How do I model multiple grids?	4-3
4.1.5	How do I model a dike-breach using the 1D channel flow module?	4-5
4.1.6	How to interpret the 'Simulation messages'	4-6
4.2	GIS-related questions	4-7
4.2.1	How do I import a *.shp-file as (part of) the network?	4-7
4.2.2	How do I edit the 2D grid?	4-7
4.2.3	How do I model roads and railway dikes?	4-7
4.2.4	How do I delete a grid?	4-7
4.3	Scale-related questions	4-8
4.3.1	Which gridsize should I use in the schematisation?	4-8
4.3.2	What is the maximum possible number of grid elements?	4-8
4.4	Installation-related questions	4-9
5	Tutorial Delft-1D2D (SOBEK-Channel Flow and Overland Flow modules) ..	5-1
5.1	Setting up of the combined system	5-1
5.2	Setting up the 1D network	5-4
5.3	Setting up the 2D grid	5-5
5.3.1	2D grid	5-5
5.3.2	2D - Boundary	5-13
5.3.3	2D - Boundary corner / 2D - Line Boundary	5-13
5.3.4	2D - Breaking dam	5-14
5.3.5	2D - Initial Water Level Point	5-15
5.3.6	2D - History Node	5-15
5.4	Simulations	5-16
5.5	Results	5-16
5.5.1	Results in Maps	5-16
5.5.2	Results in Charts	5-16

Appendices

A Example grid file (test1.asc)

B Background information on 1D Dam Break reach

I Introduction

I.1 General

In January 1998, RWS | DWW and WL | Delft Hydraulics reached an agreement to work together on a project to combine the functions of Delft-FLS and the SOBEK channel flow module. The goal was to create a single computer model that could be used to model combined 1D and 2D flow for different scenarios, like for example a dike-breach. The agreement resulted in a number of projects which had as a goal to improve and expand the possibilities of the program. Many refinements were added, among which the concept of multiple grids.

The resulting computer model is known as 'Delft-1D2D', or, within the SOBEK framework, the 1D channel flow module and the 2D overland flow module.

This manual is one of the results of the latest phase of project 'Delft-1D2D development' (Q2755), and it contains the user manual for the SOBEK overland flow module and a tutorial. This manual is delivered together with version 2.30.000.10 of Delft-1D2D.

I.2 Outline of this report

Chapter one starts with the introduction. Chapter two continues with explaining some of the most important principles of the link between the 1D channel flow module and the 2D overland flow module. This knowledge is necessary for the user in order to start off making a good 1D2D schematisation.

Chapter three gives the user a basic insight into the use of the user-interface of SOBEK with respect to the additions that have been made for the 2D module. This manual can not be separated from the standard SOBEK user manual. This means that the inexperienced SOBEK user will need both manuals!

Chapter four, the Frequently Asked Questions, supplies the user with additional useful information on a number of important topics, for example the best way to model certain phenomena like a culvert or a dam-break.

Finally, in chapter five, a tutorial is presented that gives the user the opportunity to try a first schematisation for him/herself. This tutorial is also very helpful because it summarizes much of the information given in the earlier chapters.

1.3 Definitions

Word	Description
ArcView	ArcView is a desktop-GIS application, that gives the user the possibility to solve GIS related problems in a very user-friendly environment.
CMT	Case Management Tool. A tool from the Delft-Tools family used as a framework, within which different tasks can be carried out. All file management actions between the tasks are carried out automatically by the CMT.
Delft-1D2D	Combined modelling system of SOBEK Channel Flow and Overland flow module.
Delft-FLS	Purely 2D hydrodynamical flood modelling system.
DEM	Digital Elevation Model. A DEM is a representation of terrain heights in grid format.
GIS	Geographic Information System. Geographically oriented database used to analyze and present spatially distributed data.
Netter	A tool from the Delft-Tools family. It is used within SOBEK to view and edit schematisations, and to visualize results from simulations.
SOBEK	A framework of different modules developed to model both water quantity and quality processes in (mainly) urban, rural and river systems. The name comes from the well-known Egyptian god of the crocodiles. Developed by WL Delft Hydraulics in cooperation with RIZA, DHV and RWS.

Note: all units used in this report and in SOBEK are metric (S.I. standard).

1.4 Height or Depth?

In general, within the SOBEK framework, water levels, bed levels or terrain levels (heights) are defined as positive when the level is higher than a certain reference level (i.e. the mean sea level).

However, in the 2D overland flow module, the 2D terrain levels are defined as positive in downward direction. So, when providing the topographical 2D terrain level data, the values must actually be provided as 'depths', which are positive when the terrain level is lying below the reference level. The other levels are defined according to the SOBEK standard.

When you are looking at the results of a simulation in results in maps, and you happen to have both the waterlevels AND the Z-data (terrain levels) active, the scales of these two variables will be opposite!

2 Background principles

2.1 Basic Equations used for 2D computation.

The following basic equations are used for the 2D computations.

$$\frac{\partial \zeta}{\partial t} + \frac{\partial(uh)}{\partial x} + \frac{\partial(vh)}{\partial y} = 0 \quad (1)$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + g \frac{\partial \zeta}{\partial x} + g \frac{u|V|}{C^2 h} + au|u| = 0 \quad (2)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + g \frac{\partial \zeta}{\partial y} + g \frac{v|V|}{C^2 h} + av|v| = 0 \quad (3)$$

where the following notation is used:

u	velocity in x-direction (m/s)
v	velocity in y-direction (m/s)
V	velocity: $V = \sqrt{u^2 + v^2}$
ζ	water level above plane of reference (m)
C	Chezy coefficient ($\sqrt{\text{m/s}}$)
d	depth below plane of reference (m)
h	total water depth: $\zeta + d$ (m)
a	wall friction coefficient (1/m)

These equations describe unsteady two-dimensional shallow water flow.

The equation (1), the so-called continuity equation, ensures the conservation of fluid.

The equations (2) and (3) are the momentum equations. They consist of acceleration terms, the horizontal pressure gradient terms, advective terms, bottom friction terms and wall friction terms. These equations are non-linear and they are a subset of the well-known shallow water equations, that describe water motion for which vertical accelerations are small compared to horizontal accelerations (this applies to tidal flow, river flow, flood flow).

As opposed to the shallow water equations, equations 1-3 do not incorporate the turbulent stress terms, accounting for the subgrid transfer of momentum in between gridcells. These terms have been omitted because they are relatively unimportant for flood flow computations, in order to save computational effort.

The wall friction terms have been introduced to account for the added resistance that is caused by vertical obstacles, like houses or trees. The wall friction coefficient is based on the average number and diameter of the obstacles per unit area and the average obstacle drag coefficient (C_d coefficient).

2.2 Equation Solving Method

The equations are solved by a so-called “Delft Scheme”. The solution method has been specifically designed to ensure positive solutions, i.e. negative water depths cannot be a result of the computations. Traditional solution methods do not ensure this property and therefore need special control structures to tackle the computation of waterlevels at shallow levels (the so called 'drying and flooding' procedures). The present solution technique is capable to compute bores, hydraulic jumps, supercritical flow and overland flow all in one code.

To use this scheme, it is assumed that each grid cell is a node, connected to the adjacent cells by four branches (see Figure 2.1).

2.3 1D-2D Connection

The 1D network is linked with the 2D grid in the following ways:

1. The connection between the 1D connection Node and 2D grid cell;
2. The connection between the 1D calculation Points and 2D grid cell.

The following rules should be kept in mind, only one connection per grid cell is allowed. In other words, you cannot have both a connection node and calculation point in one grid cell, nor more than one connection node or calculation point per grid cell. It is simpler to assume that 1D and 2D networks are two independent map layers, with the 2D network map layer overlapping a 1D network. The computational code determines the connection points between 1D and 2D based on the map coordinates for the center of 2D grid cell and the 1D connection node / calculation node. If they fall within certain criteria, then the connection is made between them, else not. The criteria, if expressed in mathematical terms, are as follows:

$$\text{if } (|X1-X2| \leq DX/2) \text{ and } (|Y1-Y2| \leq DY/2),$$

where: $X1$ = x map coordinate for 1D point

$X2$ = x map coordinate for 2D grid cell

$Y1$ = y map coordinate for 1D point

$Y2$ = y map coordinate for 2D grid cell

DX = width of grid cell in X direction

DY = width of grid cell in Y direction (DX and DY are equal)

then the 1D point is assumed to lie completely within the 2D grid cell.

The connection between the 2D cells and the 1D network is done in the following way (see Figure 2.1):

1. The center of 1D node is internally moved to match with the center of 2D grid cell, without changing the length of the connecting 1D branches.
2. The 2D Grid Cell is counted as part of 1D Node.
3. The flow in 1D channel below the 2D grid level is treated as 1D flow, while the flow above the 2D Grid level is treated as 2D flow with the area of 2D grid cell.

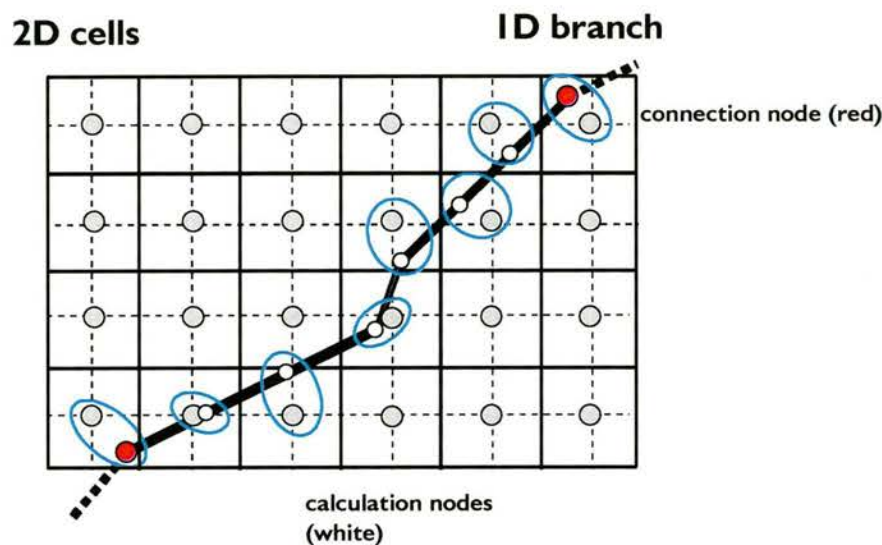


Figure 2.1 Connections between 1D en 2D

2.4 2D-2D Connection

The 2D grid cell connection to another 2D grid cell when one grid is partially or fully overlapping another grid, is done in such a way that the 2D Grid cell of the nested grid is connected to the 2D grid cell of underlaying grid by a branch. This branch carries the flow to and from the two connecting grid cells. This branch is given the property of the smallest grid cell and with a very low friction coefficient so that the hindrance of flow is negligible.

For more information on the different kinds of multiple grid configurations (i.e. parent/ child grids) possible, please refer to the Frequently Asked Questions section.

The computational code defines the connection between two 2D grid cells based on the map coordinates for the center of each grid cells. If they fall within certain criteria, then the connection is made between them else not. The criterias, expressed in mathematical terms, are:

$$(|X1-X2| \leq DX) \text{ and } (|Y1-Y2| < YH \quad \dots 1)$$

$$(|Y1-Y2| \leq DY) \text{ and } (|X1-X2| < XH \quad \dots 2)$$

where $X1$ = x map coordinate for 2D grid cell in first grid
 $X2$ = x map coordinate for 2D grid cell in second grid
 $Y1$ = y map coordinate for 2D grid cell in first grid
 $Y2$ = y map coordinate for 2D grid cell in second grid
 $DX = (DX1/2 + DX2/2)$
 $DY = (DY1/2 + DY2/2)$
 XH = maximum of $DX1/2$ or $DX2/2$
 YH = maximum of $DY1/2$ or $DY2/2$
 $DX1$ = width of grid cell in X direction in first grid
 $DY1$ = width of grid cell in Y direction in first grid
 $DX2$ = width of grid cell in X direction in second grid
 $DY2$ = width of grid cell in Y direction in second grid

The connection is made, if any of the above two criteria matches. It is possible that there are more connections made to the same 2D grid cell from different 2D grid cells in another grid.

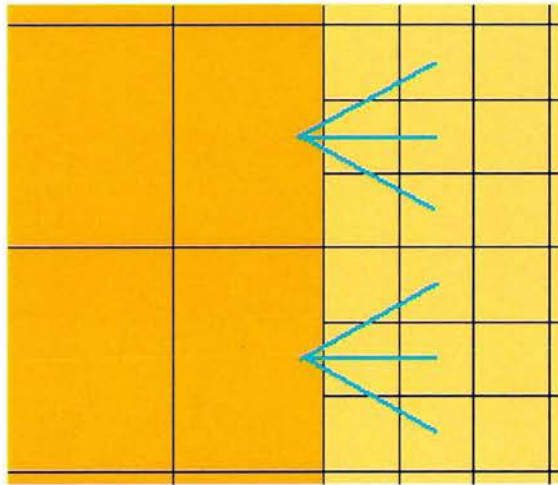


Figure 2.2 Connections made between different grids

Note: please bear in mind that the outside cells of any grid are removed from the calculation. So, in the example of Figure 2.2, the connections are made between the large (parent) grid cell and the second column of grid cells of the child grid!

3 Description of every task block

3.1 Task block I: 'settings'

3.1.1 General

The *settings* task block gives the user the following two options (see Figure 3.1):

- The combination of SOBEK modules used for the simulation can be selected here, and
- For every module selected general settings can be defined.

At this moment, there is only one possible configuration of SOBEK modules when modelling 2D-overland flow: the *channel flow* module and the *overland flow* module. This means that it's not possible to run the overland module in combination with, for example, the Real Time Control module or the Rainfall Runoff module. Even if you do not need the 1D (channel flow) module, you still need to select it, as Delft 1D2D cannot run without it. Note that in this case it is necessary to define a so called 1D *dummy branch*. This will be explained later on.

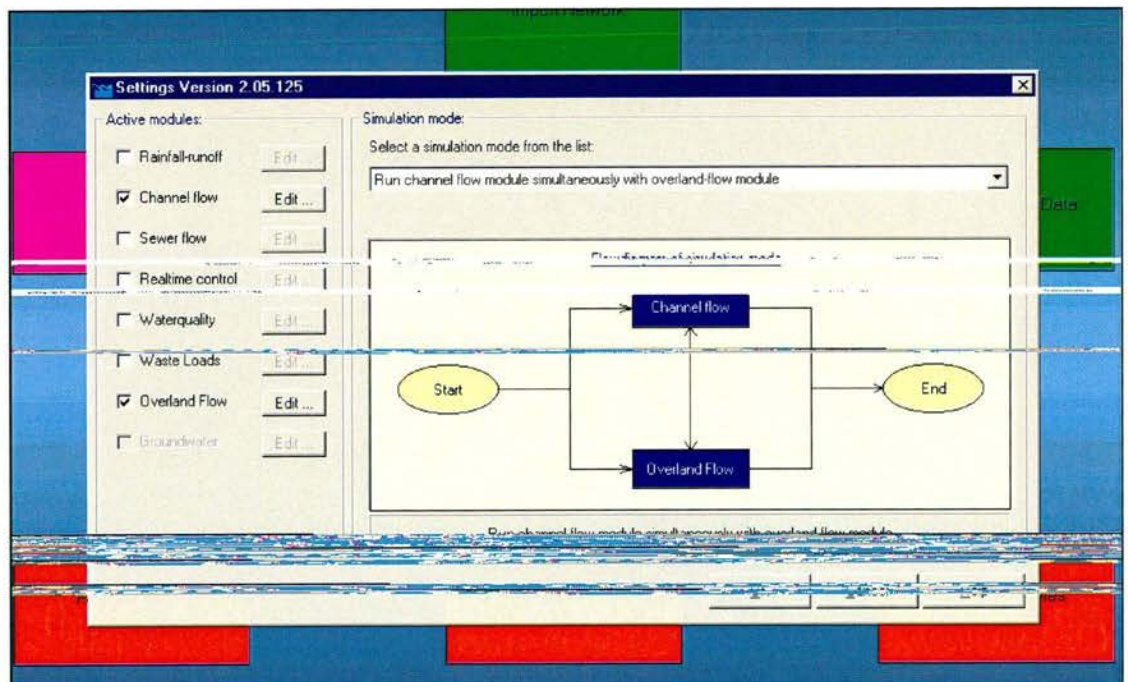


Figure 3.1 Settings

Note: Within the SOBEK family, the Delft 1D2D module is being referred to as the 'overland flow' module.

For a description of the settings of the channel flow module, please refer to the general SOBEK help documentation.

3.1.2 Overland flow settings

Settings for the overland flow module are divided into 5 different sub-sections (see Figure 3.2):

- simulation settings
- initial data
- GIS output options
- incremental output
- history output

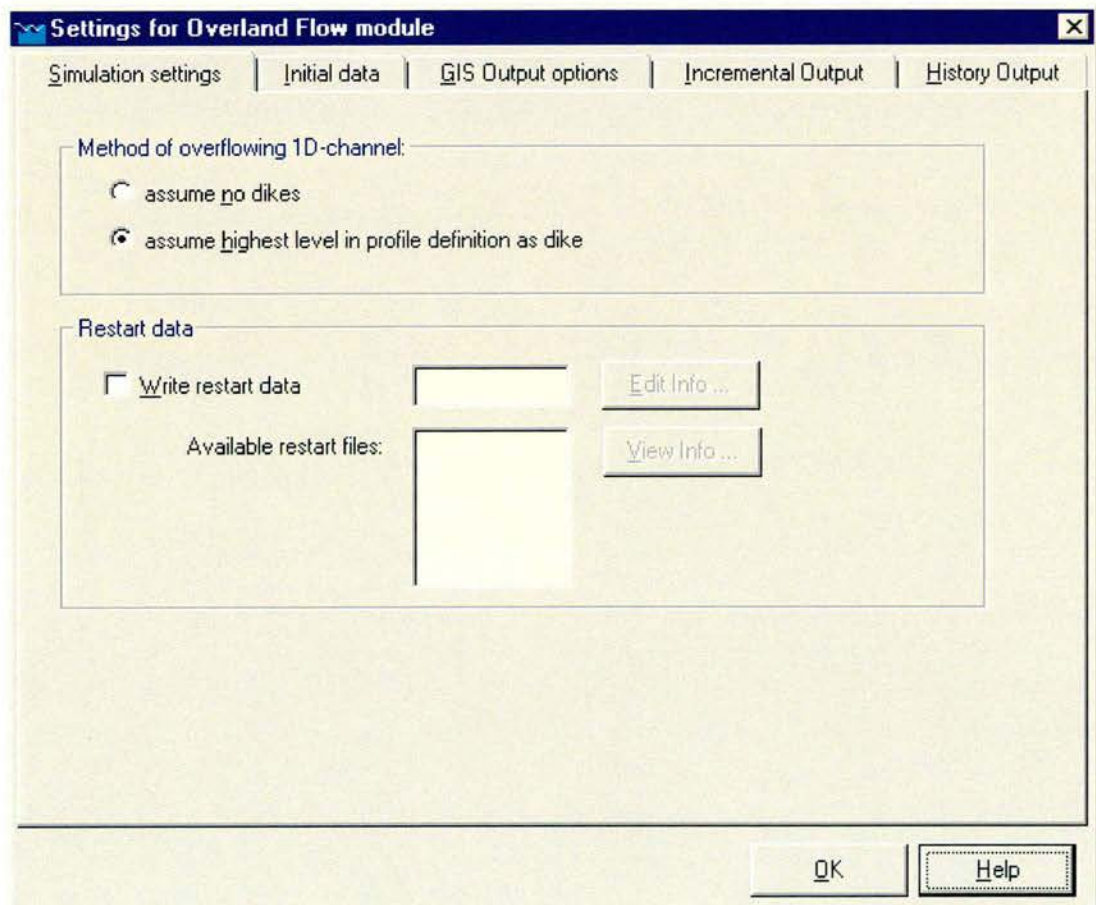


Figure 3.2 Settings for the overland flow module

Simulation settings tab

Here the user can specify a number of general options for the 2D simulation, starting with the *method of overflowing 1D channel*. This is a very important option which will greatly influence the results of the simulation, so it is very important that you set this option wisely. It concerns the connection of the 1D cross-section with the underlying 2D grid cell, and is explained in the following example (see Figure 3.3):

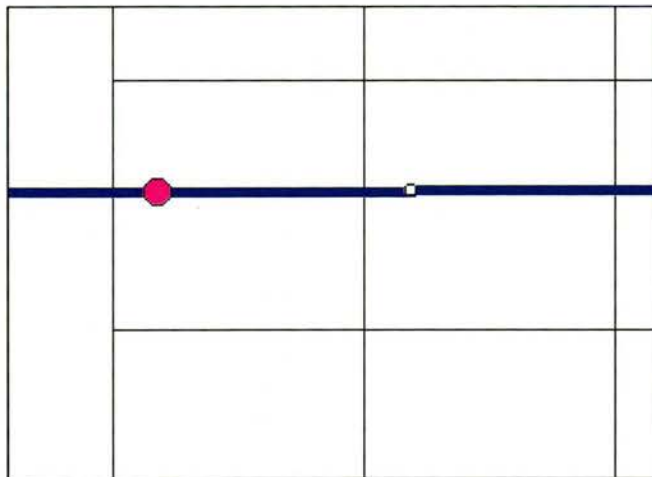


Figure 3.3 Example 1D-2D schematisation

A 1D channel delimited by dikes passes through a low lying area, say a polder. When the waterlevel inside the 1D channel overtops the dikes, water will start to flow into the underlying 2D grid elements. In Delft-1D2D, the dikes of the 1D channel can be modelled in two very different ways, as is shown in the following two cross-sections of the schematisation of Figure 3.3.



In this case, water enters the 2D grid when the highest level of the 1D profile is overtopped, and not earlier. So, the dikes form a barrier between the channel and the 2D area. Overmore, they form a barrier for water flowing towards this 1D crosssection over the 2D grid.



Notice: At this moment, the option of method of overflowing 1D-channel refers to *all* 1D crosssections.

In this case, water enters the 2D grid as soon as the channel water level reaches the terrain level in the 2D grid, so the part of the dike in the 1D cross-section above this level is neglected. Please refer to the list of Frequently Asked Questions for more details on this switch.

The second simulation option concerns the writing of a *restart file*. This is a datafile which contains all flow conditions of the 2D system at the end of the simulation (i.e. the waterlevels and velocities in all grid cells). This datafile can then later be used for the initial conditions, instead of the standard conditions. **Make sure you set the same restart option in the channel flow module !!**

Initial data tab

Settings for Overland Flow module

Simulation settings | Initial data | GIS Output options | Incremental Output | History Output

Initial values

☐ define global values:

☐ initial waterlevel [m + ref.level] 0

☐ initial depth [m] 0

☒ define local initial values per 2D-grid in <Edit Network>

☐ use restart file

Selected: [dropdown] View Info ...

OK Help

Figure 3.4 Initial data tab

Note: The first option, *define global values*, cannot be selected at the present.

At this moment, there are either one or two options available here. If you haven't run a simulation before with a restart file as output, you won't be able to select the second option *use restart file*, so you will need to use the default option, *define local initial values per 2D-grid*.

There are a number of things you need to remember when using the restart option:

- make sure you select the *use restart file* option **in both the overland flow module AND the channel flow module !!**
- The restart file can only be used as input for exactly the same schematisation that was used in the simulation that generated the restart file.
- The restart file itself is saved under the project directory; this means that you will only be able to use this restart file as input for a (any) case within the same project, but not for a case in another project.

GIS output options tab

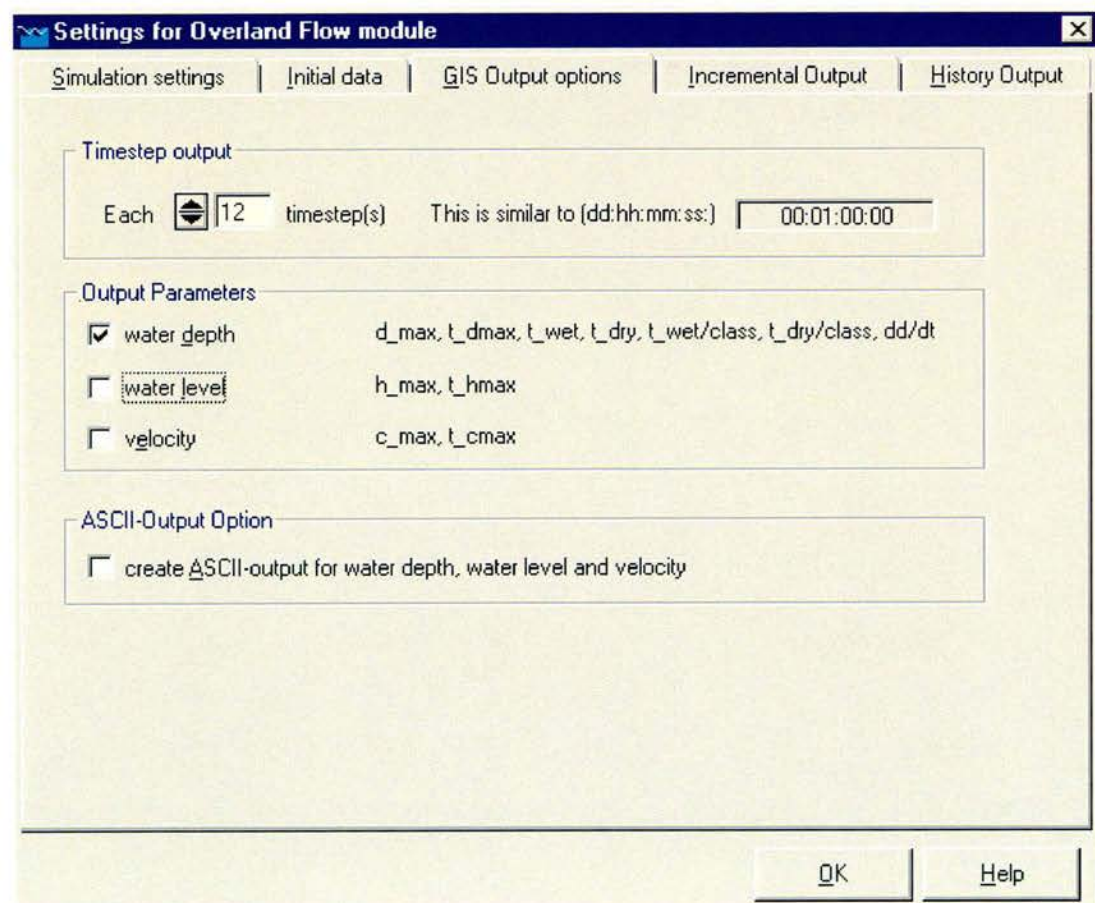


Figure 3.5 GIS output tab

The output for GIS systems, like ARCVIEW, is one of the three types of 2D-specific output available. In this screen, you can select if you want this output to be generated or not, and for which timesteps. If any (combination) of the available variables is selected here, SOBEK will automatically create '.ASC' output files which can be imported into a GIS based program. There are some additional rules to remember, which will be explained further on.

The GIS output data is written to a number of different files. The first type consists of output files in .asc format (see appendix A). SOBEK creates these output files only when the switch 'create ASCII-output for water depth, water level and velocity' is switched on! One of these is generated for every selected output timestep and for every selected variable. In general, it is a good idea to wait with generating this output (if you need it at all, of course) until you are pretty sure that the simulation is going to give you the results that you need. Suppose your simulation lasts 24 hours, and you want GIS output for every timestep (of, for example, one minute) and for all five variables, you would get 5x24x60 equals 7200 output files!

Warning: Right now, SOBEK can handle only a certain maximum number of files in the case directory. If you decide to switch on the 'ASCII-Output Option', you might actually reach this maximum number of files (because of all the .asc files generated), which means that you might not be able to save your case properly! In that case you will get the message with saving your case: *“too many files in case directory; Check the configuration of the application; erase unregistered files”*. In this case you should reconsider you output choices to decrease the number of output files.

The second type of output files is the binary '.map' file. For every variable chosen, a file is produced. It contains the values of that variable for every timestep and every grid element of one particular grid. In *results in maps*, the user can select these .map files and produce .asc files out of them. So actually, the .map file contains the same information as the .asc files described earlier, but in a compressed format. Please note that whether MAP files are written or not, only depends on the *Output Parameters* chosen in the *GIS Output Options* tab, and not on the *ASCII-Output option* switch!

The last type of GIS output files are 'special' .asc files. These files contain the following variables:

- maximum water depth (d_max)
- time of maximum water depth (t_dmax)
- maximum water level (h_max)
- time of maximum water level (t_hmax)
- maximum velocity (v_max)
- time of maximum velocity (t_vmax)
- time of wetting (t_wet)
- time of wetting per class ('permanentie')(t_wet/class)
- time of drying (t_dry)
- time of drying per class ('permanentie') (t_dry/class)
- rate of change of water depth (dd/dt)

These files are only produced when the corresponding 'basic' variable is selected in the GIS tab, see figure 3.5. For example, the maximum depth is generated only when the option 'water depth' is checked. The same goes for other variables like the maximum velocities (linked to 'velocity') or maximum levels (linked to 'water level'). The files giving a time (period) and the rate of change of water depth are all linked to the 'water depth' output option. Please refer to paragraph 3.3.4 for more information about these files and their filenames.

Incremental Output tab

Settings for Overland Flow module

Simulation settings | Initial data | GIS Output options | **Incremental Output** | History Output

Timestep output (same as Channel Flow)

Each timestep(s) This is similar to (dd:hh:mm:ss:)

Output Parameters

- ☒ water depth
- ☐ water level
- ☒ velocity
- ☒ velocity(u component)
- ☒ velocity(v component)

Classes for incremental file:

Parameter:

Number of classes:

H <	-0.01
-0.01 < H <	0.1
0.1 < H <	0.2
0.2 < H <	0.3
0.3 < H <	0.4
0.4 < H <	0.5
0.5 < H <	0.6
0.6 < H <	0.7
0.7 < H <	0.8
0.8 < H <	0.9
0.9 < H <	1

OK Help

Figure 3.6 Incremental output tab

The second type of 2D output is the so-called 'incremental output'. This is output that is generated specifically for result analysis in SOBEK ('results in maps') itself, so you should normally have at least some of these options turned on. Otherwise, you will not be able to visualize the results in SOBEK.

The incremental file system was developed to reduce the enormous amount of data generated by the 2D simulations. It works by defining a number of classes for every output variable which are then used in the output files instead of the actual data value itself. So, for example, if a waterdepth in a certain 2d grid cell equals 0.43 m, it would fall in class '0.4 - 0.5' (if you have specified such a class). This is also the result you would get when

examining the results for that particular grid cell in 'results in maps'. So, this output is especially useful for getting a quick idea as to how the water flows through the 2d grid system; if you are interested in the exact values for the variables for a certain number of gridcells, there is another option available to the user, which is the *history output*, explained in the next paragraph.

The incremental output always has the same timestep as the actual calculation timestep chosen in the *channel flow* module.

Concerning the output parameters, it is important to know that the *velocity (u component)* and the *velocity (v component)* are necessary if you are interested in seeing the *velocity vectors* later on, when viewing the results in *results in maps*.

The number of classes can be selected manually, and influences the detail you will get when viewing the results; ten classes of 10 cm difference show much more information than 2 classes of 50 cm difference, but also require more disk space and computation time. The number of classes will be the same for all variables, so it's not possible to have 5 classes for the waterdepth and 15 for the velocity. But, if you have defined 15 classes (for all variables), and you need only 5 for one particular variable, you can define the unnecessary classes as no-data values (-999).

A few rules to remember:

- The waterdepth can be only positive, so it's no use to define negative classes here
- The 'u' and 'v' velocity components can be both positive and negative. It's important to remember to use exactly the same classification for both variables.
- This classification needs to be classified *symmetrical* around zero for correct determination of the vectors!

***History Output* tab**

If you are interested in the exact values of all variables on specific locations in the 2D grid, it is possible to use a special kind of node called a *History station*. How to do this is explained in the chapter three. In this tab you can define the timestep for this output.

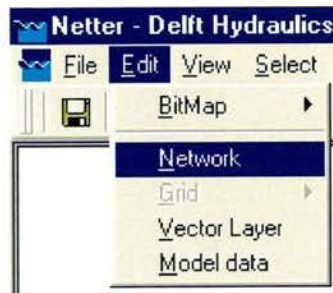
3.2 Task block 2: 'schematisation', Edit model

3.2.1 General

Within this task block the actual schematisation is built and verified. As mentioned before, this manual will only deal with the 2D additions to the 1D *channel flow* version of SOBEK. Please refer to the general SOBEK user manuals for more details on the outline of schematisation task block and the use of the 1D channel flow module in specific.

Just like the 1D schematisation, the 2D schematisation is built in two steps using a number of building blocks. In the first step, the *edit network* mode, the grids and the other building blocks, like initial water level nodes, are being placed on the map. In the second step, the *edit model data* mode, the characteristics of all these building blocks are defined. Both steps will be explained in detail in the following two paragraphs.

3.2.2 Edit network mode



Introduction

This paragraph starts with giving a summary of a number of important aspects when making a 1D-2D schematisation. After that, all types of 2D nodes are explained, and what rules should be obeyed when placing them on the map. Paragraph 3.2.3, 'Model data mode', will explain about all the model data that needs to be specified for the 2D nodes.

General aspects of a 1D-2D model

Figure 3.7 depicts the currently available building blocks (called 'nodes' in SOBEK). For now, two groups of nodes are available, namely the flow nodes and the 2D nodes. The flow nodes are the ones used to build 1D schematisations, please refer to the SOBEK Channel-flow manual. The 2D nodes are the elements that have been added to SOBEK for the modelling of 2D systems. Together they make up the 1D-2D schematisation.

One important 2D building block that is not included in Figure 3.7 is the **flow - dam break** branch. This is a branch type, and not a node, which means that it can be found with the branches.

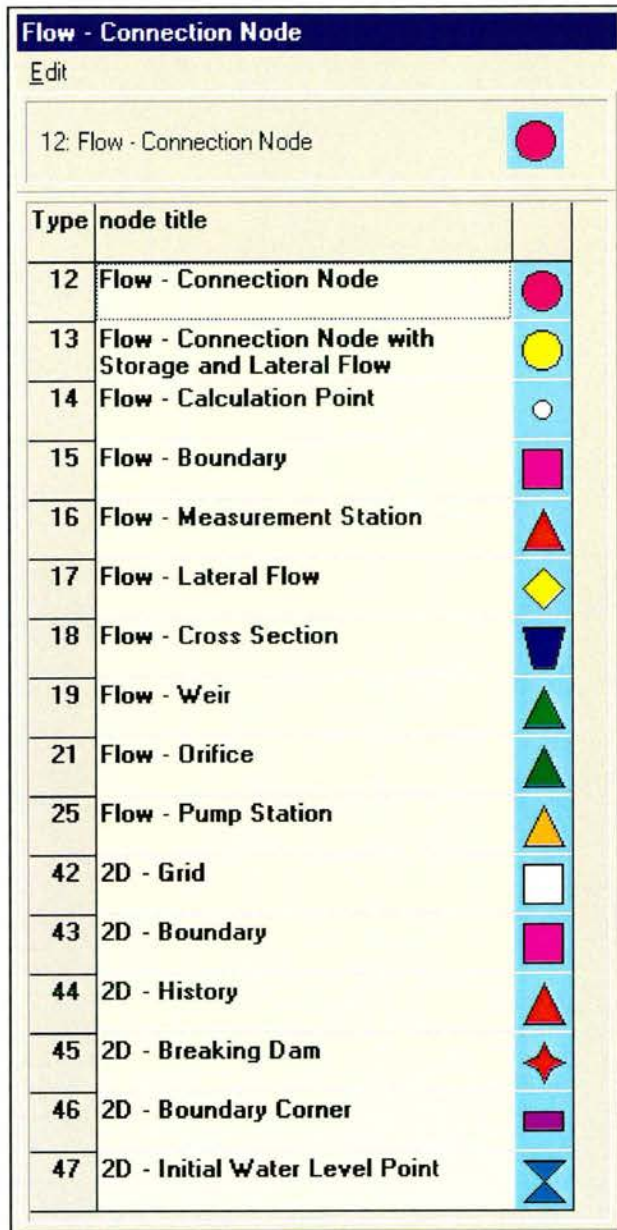


Figure 3.7 Building blocks

A very important aspect to consider when using this type of schematisation is the interaction between the 1D and the 2D system. The theory behind this interaction was explained in chapter 2, and the FAQ-section contains an example of a dike-breach modelled in 1D.

When building a 2D schematisation, there are a number of rules that need to be obeyed. One of the first rules is that every 2D-schematisation needs to contain at least one 1D-branch, even when the schematisation is purely 2D. In this case, the 1D-branch is referred to as a 'dummy-branch'. Usually, this branch would be placed somewhere outside of the grid, and its properties would be as simple as possible. Figure 3.8 shows an example:

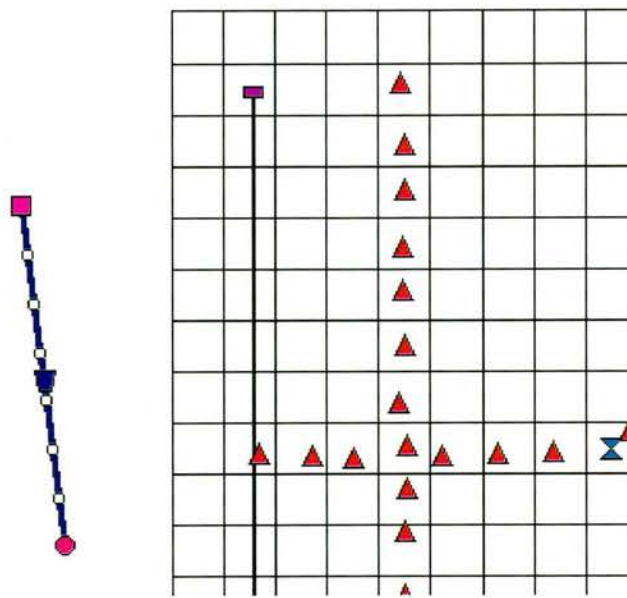


Figure 3.8 Example of 1D dummy branch

The second rule is that it is not possible to put more than one object in one 2D grid cell. The only exception to this rule is the 2D- history node, which can be combined with any other type of 2D node.

One also needs to take care when combining 1D and 2D schematisations. Some combinations of elements will not be possible because of the (automatic) links made between the two models. As explained in chapter 2, links with the 2D grid are only possible starting from a 1D connection node or a 1D calculation point, and not from any other type of 1D node, like a 1D boundary. Because there can be only one 1D node connected to a 2D grid cell, it is a good idea to have exactly One, no more and no less, 1D calculation point (or connection node) defined per 2D grid cell.

Finally, the last important fact to remember is that SOBEK internally removes all outside grid cells from the grid, before starting the simulation! So the grid used for calculation will be 2 columns and 2 rows smaller. Knowing this, it is also not possible to define any 2D nodes in any of the outer grid cells. If you do so anyway, SOBEK can crash during a simulation.

2D - grid node



This is one of the most important building blocks for the modelling of 2D systems, and the one you would usually start with. It is referred to in the user interface as a 'node' which at first sounds confusing, as this node actually represents a whole grid.

A grid can be either imported from a GIS system, or defined within SOBEK as a new grid.

Importing a '2D grid' is possible by selecting the 'import 2D grid' option, an option which has been added to the standard SOBEK *edit network menu* (see Figure 3.9). Note that this option is only available after you have selected the '2D grid' node as a building block.

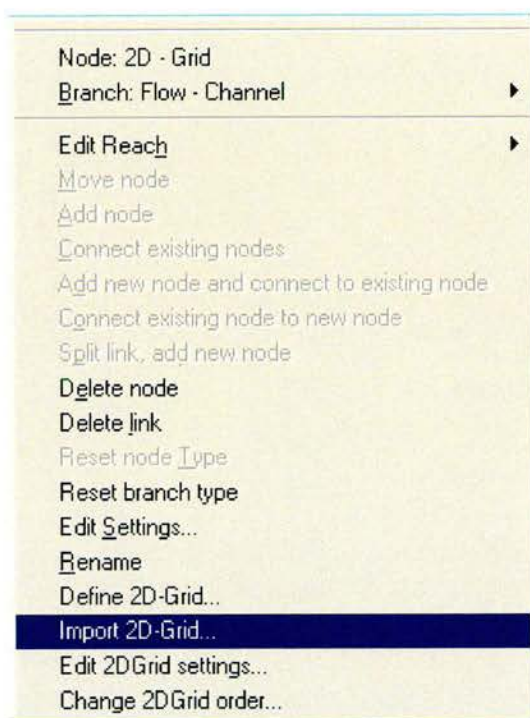


Figure 3.9 Edit network menu

After you have selected this option, you need to select the .asc file that contains the grid information you want import. As mentioned before, this .asc file is a standard grid definition file, which can be generated as output by for example ARCVIEW (see appendix for an example). The file contains the following information about the grid:

- the number of columns
- the number of rows
- x-coordinate of the bottom left corner of the grid
- y-coordinate of the bottom left corner of the grid
- cellsize of the grid elements (same for x and y size)

- the no_data value ('missing value'), usually -999 or -9999
- For every cell the terrain level **BELOW** a certain reference level (depths).

In theory, you can select the file from any given path, for example `c:\gisfiles\grid1.asc`. However, we strongly recommend using the default directory for these filenames, which is the `<projectname>\fixed\` directory. This makes it easier to make a copy of an existing project and give it to somebody else.

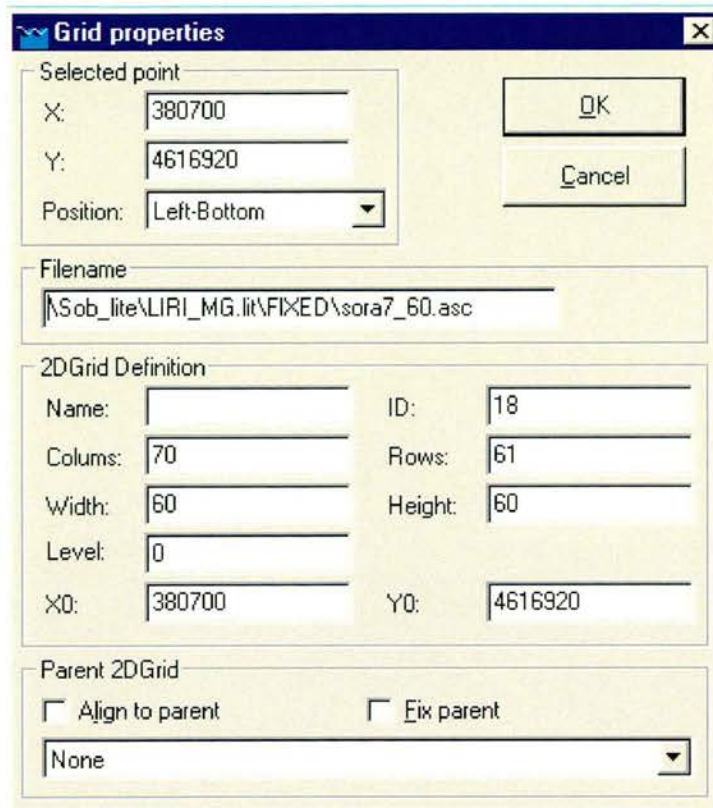
After you have successfully imported the grid, it should appear on the map at the coordinates specified in the file. You can then visualise the terrain levels of all gridcells by turning on the *active legend* (under *options*) and activate the *z-data (model data)* from the legend panel. The *active legend* will be explained later on.

Most of the time, grids will (need to) be imported from GIS based programs. It is however also possible to **define the 2D grids** by hand, by selecting the 'define 2D grid' option from the *edit network menu* (see Figure 3.10). This means that the user will not only have to define the exact location and size of the grid, but also the terrain level for every grid element. In theory, one can place the grid at any given location and make it any given size. However, the way the grids are defined have a major impact on the accuracy of the results, so accurate definition of 2D grids is very important. There are a number of considerations important in this matter:

- The number of grids and the number of elements determine the simulation time. The less elements there are, the shorter the simulation takes.
- The smaller the grid elements are, the more accurate the results can be. It is therefore a good idea to choose a coarse grid in (most) places where accuracy is not required, and a more fine grid in places where results are critical, for example close to the inlet point/dam break point. Another option is to choose the resolution of the grid in the same order as the resolution of the available terrain level input data.
- The grid elements containing a **no-data value** don't participate in the calculations; they are used to specify the barrier around the area of interest, and possibly as no-flow locations within the area of interest.

Grid properties

For every grid you need to specify a number of properties, some of which need to be specified immediately, while others can be defined later in the 'model data' mode. Figure 3.10 shows the properties that need to be defined.



Grid properties

Selected point

X: 380700

Y: 4616920

Position: Left-Bottom

OK

Cancel

Filename

\\Sob_lite\LIRI_MG.lit\FIXED\sora7_60.asc

2DGrid Definition

Name: ID: 18

Columns: 70 Rows: 61

Width: 60 Height: 60

Level: 0

X0: 380700 Y0: 4616920

Parent 2DGrid

☐ Align to parent ☐ Fix parent

None

Figure 3.10 Grid properties

Selected point information

The 'X' and 'Y' coordinates are the coordinates of the map position you selected by pressing the mouse button just before entering this menu. They represent a (any) corner of the 2D grid you are about to define. If you are not satisfied with the coordinates, you can alter them manually here.

The *Position* represents which of the four corners should appear at the X-Y coordinate you defined. By default it's the left-bottom corner.

Filename

The default location for putting the .asc files is the <project directory>\fixed\ directory. Make sure you select a path before saving the grid file.

2D grid definition

- **Name**
If you want to, you can specify a unique grid name. This is not absolutely necessary, as a default name will be defined for you if you don't specify a name here.
- **Columns / Rows**
Every grid is rectangular in shape, the size of which is determined by the number of *rows* and *columns*. The area of interest is inside the rectangular grid, which is filled up with the *no-data values*.

- Width / height (m.)
The width and height of every cell are uniform for the whole grid, so it's not possible to specify varying grid size elements throughout one grid. However, different grids within one schematisation can have different grid sizes.
- Level (m. below reference level)
When using the *define 2d grid* option, it is only possible to define one uniform terrain level. Later on, in the *edit model data* mode, there is the possibility to change terrain levels for single grid elements. One important thing to remember: the *level* is defined as **positive** in **downward** direction. In other words: the level is actually the **depth**!
- X0 / Y0
This is the same coordinate as specified under *selected point*.

Parent 2D-grid

This option is only relevant when using multiple grids that are either **overlapping** or **nested**. Please refer to the FAQ for more details on how to model multiple grids.

If you select the *align to parent* option, the child grid will be replaced according to the parent grid. If you select *Fix parent*, the position of the parent grid will be changed according to the position of the child grid. Note that the position of the top left corner of the grid selected will be changed, and not the number of rows or columns. So, make sure to check both the top left and bottom right corner of the child grid afterwards to check the alignment. It may be necessary to adjust the size (columns/ rows) of the child grid manually according to the rules specified in the FAQ.

Note: due to an error in the options 'align to parent' and 'fix parent', these options can not be used at the moment! This error will be solved in a future version of SOBEK

Edit 2D-grid...

This option is available under the *edit network* menu (see Figure 3.9). Opening the **grid properties** window (see Figure 3.10), it gives the user the possibility to change grid properties, after the grid has been defined. If you decide to resize the grid (number of columns and/ or rows, don't forget to edit the terrain levels in the .asc file accordingly.

Before you can edit the properties of any grid, you will first have to select the node that represents the grid. And in order to be able to select this type of node, they first have to be made visible. This can be done by selecting 'options' => 'network options' => 'nodes...' => select the 2D grid node and turn on visibility (see Figure 3.11). The 2D-grid nodes should now be visible in the schematisation.

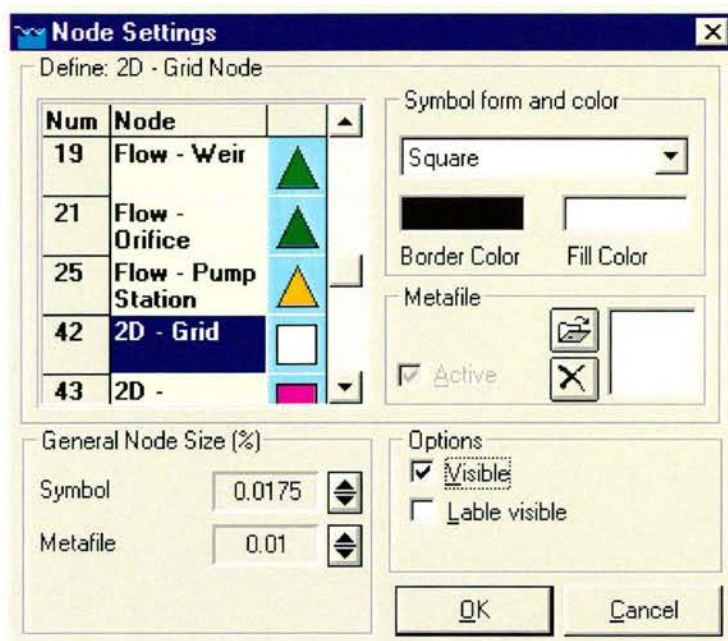


Figure 3.11 Node settings

Change 2D-grid order....

After selecting this option from the *edit network* menu, Figure 3.12 appears:

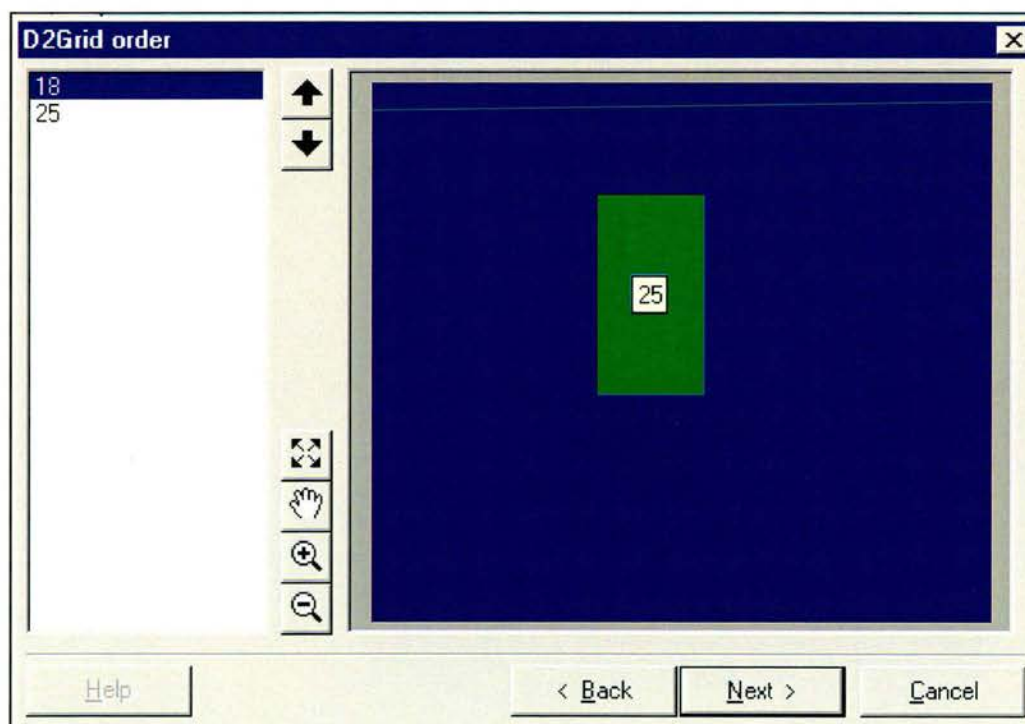


Figure 3.12 2D grid order window

Here you can change the order of nested grids according to the rules specified in the FAQ on multiple grids. In short, the child grid (in Figure 3.12 the small green grid, grid 25) should be 'on top of' the parent grid (the blue grid, grid 18).

This means that in the table on the left, the parent grid should be mentioned BEFORE the child grid. If the grid order is wrong (clearly visible in the 2D grid order window because the child grid is hidden behind the parent grid), the simulation will crash. By default, the grid order is defined by the order in which the grids are defined in the first place. So if you first define the parent grid, and then the child, the grid order will be correct. If you first define the child, and the parent, the order needs to be changed using this option.

So, it is very important to always check this option after defining new grids!

2D - boundary node



The 2D - boundary node works more or less the same as its counterpart in the 1D-channel flow module. The boundary condition is either a water level or a discharge, both of which can be either constant or varying in time.

Any 2D grid cell can discharge into any of the four cells directly surrounding it. However, a 2D cell containing a boundary can only discharge into (any) ONE of these cells! By default, it discharges into the cell directly to the right of it. The flow direction can be controlled by defining no-data values in the surrounding cells to which the boundary cell should NOT discharge. This way, if you define 3 no-data values out of 4 surrounding cells, the boundary cell will discharge into the remaining cell (see example Figure 3.13).

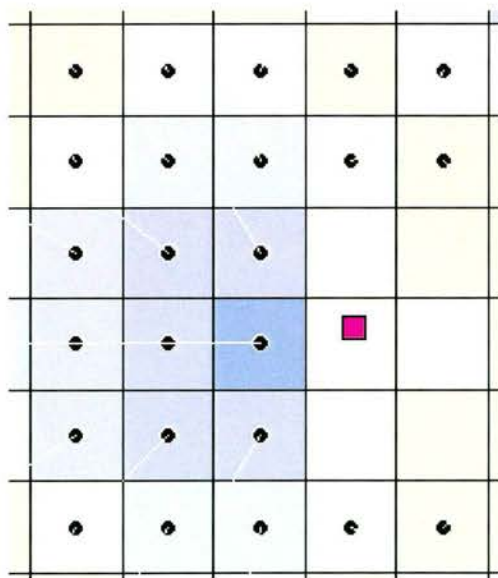


Figure 3.13 Possible configuration of 2D boundary node

Note: the white (transparent) cells contain no-data values

If you decide to use a Discharge boundary condition as part of the 2D schematisation, it is wise to define an initial water level point in the 2D grid cell into which the cell containing the boundary condition should flow. The reason for the initial water level point in this case is purely a numerical one, as sometimes no water will flow at all from the discharge boundary. The water depth can be very small, for example one centimetre.

The initial water level node will (instantly) fill up all surrounding 2D grid cells with water, until cells are reached that have a higher terrain level, or the edge of the grid is reached. This can be a problem when using a discharge boundary condition on a high point in the grid: an initial water level used for numerical reasons may fill a large part of the grid! This problem can be solved by lowering the terrain level of the 2D grid containing the initial water level point slightly below the level of the surrounding cells.

One more rule to remember when using this node is that the 2D grid cell containing the node should be defined manually as 'no-data value!'. This can be done in 'edit model data' mode, under the properties of the 2D grid.

Finally, it is not possible to add this type of node to a grid cell that already contains a 1D-boundary node.

2D-Boundary nodes are normally used in situations where the dambreak is modelled starting in the 2D schematisation, as opposite to starting in the 1D-schematisation.

2D - history node

44 2D - History



As explained in the paragraph about the *settings* task block, the default output of the SOBEK overland flow module has the format of incremental files, which was developed to reduce the enormous amount of data generated by the 2D simulations. It works by defining a number of classes for every output variable which are then used in the output files instead of the actual data value itself. So, for example, if a waterdepth in a certain 2D grid cell equals 0.43 m, it would fall in class '0.4 - 0.5'. This is also the result you would get when examining the results for that particular grid cell in 'results in maps'. So, this output is especially useful for getting a quick idea as to how the water flows through the 2D grid system. The incremental output is available for all selected parameters (i.e. waterdepth, velocity) and for every 2D grid cell in the schematisation.

The *2D-history* node is a special type of object which can be used to specify 2D grid cells where the incremental output is not enough and exact output is required. After the simulation has finished, the additional output will be available as 'results at history stations' in Netter.

2D - Breaking Dam node

45 2D - Breaking Dam



Dike breaches can be modelled either in 1D or in 2D. If you prefer to do it in 2D, you will need to use the **2D - breaking dam** node. By placing one of these nodes inside a 2D grid element, you can define the terrain level of that element as a function of time.

So, you can model the effect of a breaking dam by lowering the terrain level over a certain period of time.

Figure 3.14 gives an example of a schematisation containing two 2D breaking dam nodes.

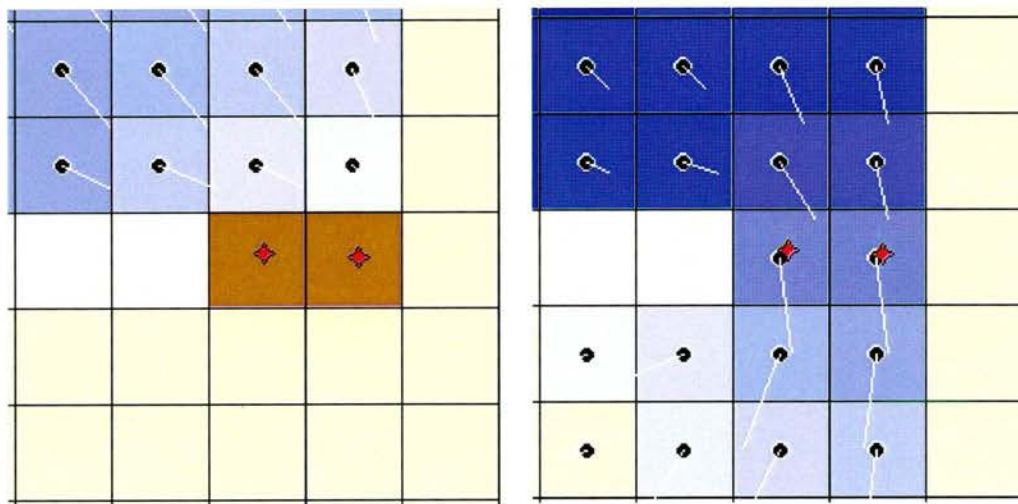


Figure 3.14 Example of breaking dam nodes

2D - Boundary Corner node / 2D - Line boundary

46 2D - Boundary Corner



The 2D-line boundary can be used instead of the 2D-boundary node when there are multiple grid cells next to each other that have the same boundary condition (i.e. tidal boundary). The boundary condition specified for the line boundary (in *edit model data* mode) is used for all underlying 2D grid cells.

The 2D-line boundary is made up of a construction of two 2D-boundary corner nodes and one 2D-line boundary. This construction is made by first placing the 2 nodes on the grid, and then connecting them using the '2D-line boundary' reach type and the 'connect existing nodes' option from the edit network menu. The line boundary can be either horizontal or vertical, but not anything else. All underlying 2D grid cells should contain no-data values, and from every boundary cell the water should be allowed to flow to (or from) only one other cell. See Figure 3.15.

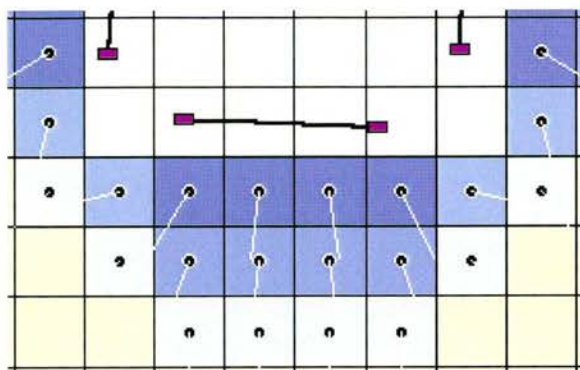


Figure 3.15 Possible configuration of a 2d - line boundary

2D - Initial Water Level Point node



This node can be used to create an initial water level in a part of the grid. This is useful in two cases:

- Initially there really is water in part of the 2D system, for example a lake.
- A discharge boundary condition is used as part of the 2D schematisation. In this case, it is best to define an initial water level point somewhere to make sure that the 2D grid cell adjacent to the cell containing the boundary condition is not initially dry. So, the reason for the initial water level point in this case is purely a numerical one, as sometimes no water will flow at all from the discharge boundary. The water depth can be very small, for example 1 cm.

The initial water level node will (instantly) fill up all surrounding 2D grid cells with water, until cells are reached that have a higher terrain level, or the edge of the grid is reached. This can be a problem when using a discharge boundary condition on a high point in the grid: an initial water level used for numerical reasons may fill a large part of the grid! This problem can be solved by lowering the terrain level of the 2D grid containing the initial water level point slightly below the level of the surrounding cells.

Flow - Dam break branch

Dike breaches can be modelled either in 1D or in 2D. The *flow - Dam break branch* gives the user a powerful tool to model them in 1D. The main advantage of this branch type over its 2D counterpart (the *2D - breaking dam node*) is that the user can actually specify the growth (both vertical and horizontal) of the breach. This will allow more accurate modelling of the flood wave through the dike breach.

The dam break branch actually behaves like a 1D weir for which both the crest level and crest width change in time. This means that the flow **through** the 1D branch is controlled, and that the dam break branch in itself does not have any connection with the 2D grid. The actual outflow from the 1D channel into the 2D schematisation, the flooding, will occur downstream of the dam break branch.

Figure 3.16 shows an example of a dam breach modelled using the *flow - dam break branch*. Here, there are two compartments separated by a dam. The dam breach is modelled using two 1D *connection nodes* and a 1D *dam break branch*. The water will flow from one connection node to the other connection node via the dam break branch (the thick green line), through the dike. In this example, the dam break branch is not connected to a 1D channel system, but establishes a connection between 2 2D grid compartments.

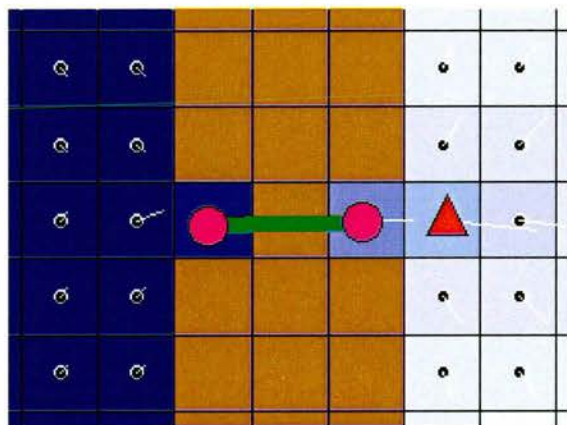


Figure 3.16 Example of a dam break branch

Figure 3.17 shows another example of a 1D *dam break branch*, in which the branch forms the connection between the 1D river system and the 2D area that will inundate. The actual flooding occurs from the 1D connection node downstream of the dam break branch. This example is similar to the 'dummy branch' example explained in the Frequently Asked Questions, paragraph 4.1.5.

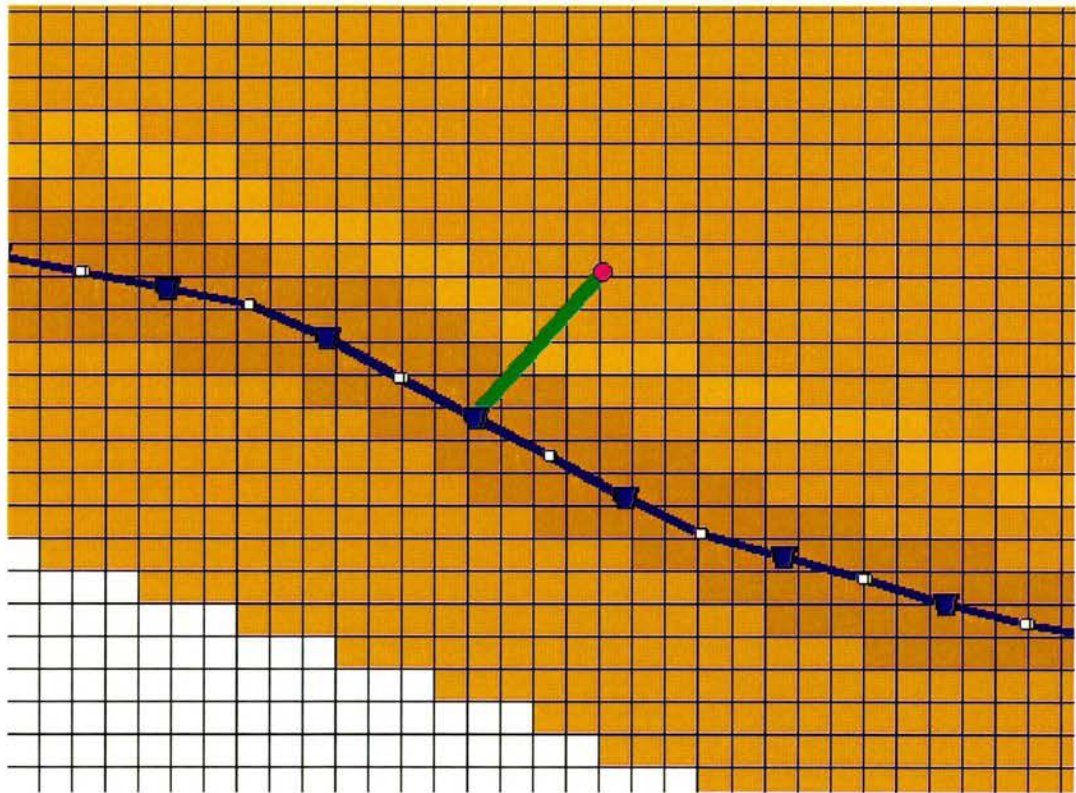


Figure 3.17 Example of dam break branch

The location of the *Flow - Dam Break* branch type is shown in Figure 3.8 Please bear in mind that this (or any other) branch can only be connected to 1D connection nodes. So the minimum configuration consists of two 1D connection nodes and the *Flow - Dam Break* branch.

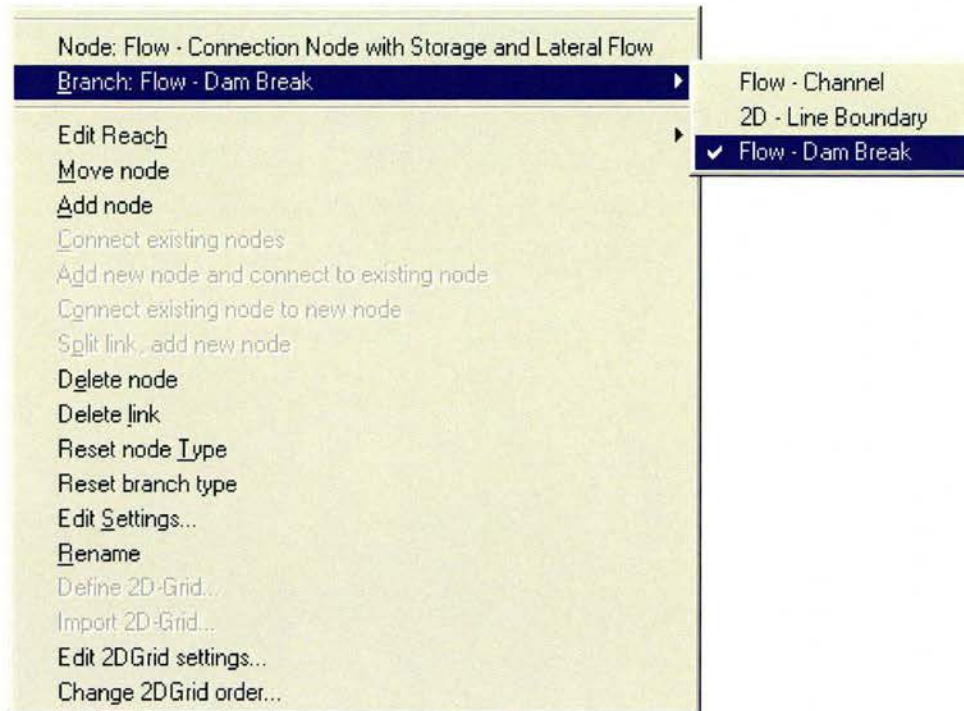
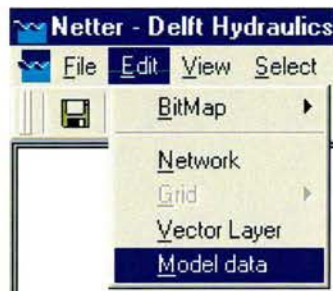


Figure 3.18 The Flow - Dam Break

3.2.3 Edit model data mode



General

The next step in building the schematisation is specifying the parameters for the different 2D elements defined in the *edit network* mode. The order in which the 2D elements are handled is the same as in the previous paragraph.

An object needs to be selected before its data can be defined. After the *edit network* option has been selected from the *edit* menu, the *model data* window appears. This window can be used to browse through all objects and select them for editing. The objects can also be selected by clicking on them in the main Netter screen.

2D - grid node



After a 2D grid node has been selected for editing, window Figure 3.19 appears:

Data for grid [X]

2D Grid Location | Grid Cell Bottom Depth | Friction | Grid Cell Friction | Defaults

Identification

ID : 16

Name :

Location

Left (X) coördinaat : 177367.5625

Left (Y) coördinaat : 480413.96875

Corner Position : 0 0 = Left Top Corner
3 = Left Bottom Center

	Number of Grid Points	Distance between Two Grid Points
in (X) Direction :	20	100
in (Y) Direction :	20	100

ARC Info File : \Sob_lite\DEMO.lit\FIXED\parent2.asc

OK Cancel Help

Figure 3.19 Grid data window - 2D grid location tab

This tab displays general information about the selected grid, which was already defined during the *edit network* mode. If you want to change any of the data specified here, you need to go back to the *edit network* mode and use the 'Edit 2D grid settings' option.

The next tab contains the terrain levels for all 2D grid elements (Figure 3.20):

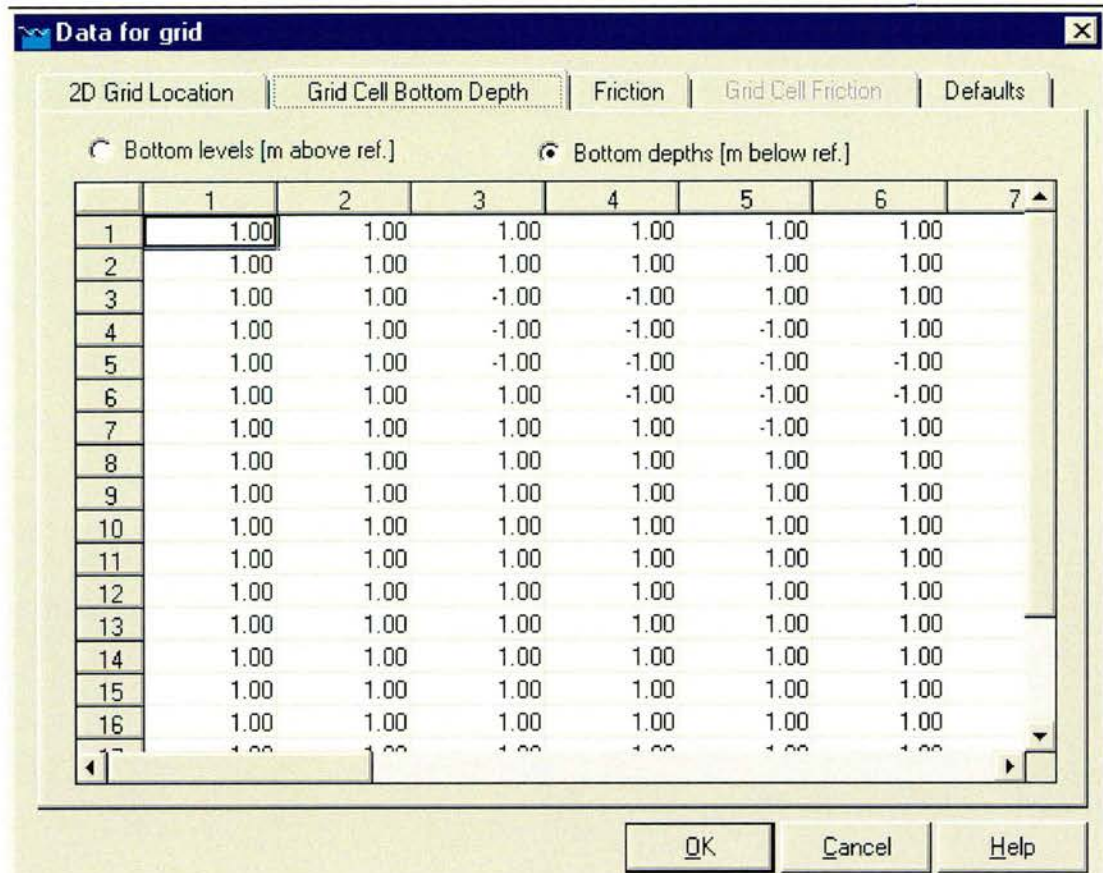


Figure 3.20 Grid data window - grid cell bottom depth tab

In this window, two aspects are of interest:

- The terrain levels in this window can be displayed as either **bottom levels (m above ref.lev.)** or **bottom depths (m below ref.lev.)**, whichever one the user prefers. The option selected here does not influence the simulation in any way, nor the way the results are displayed.
- The table contains terrain levels (either depth or height) for all grid-cells. Cell (1,1) represents the top-left corner of the grid. The user is free to alter any of these values. When the user presses <ok> after anything has been changed, the changed .ASC file containing the terrain levels can be saved under the same name or another. Please verify that the file is written to the correct directory (preferably '<casename>\fixed'), and when you enter a new filename, don't forget to add the '.asc' extension.

The next tab contains the friction values for all gridcells (Figure 3.21):

Figure 3.21 Grid data window - friction tab

The user can choose between three types of friction formulations, Chezy, Manning or White-Colebrook. Please refer to the Technical Reference Manual of the Channel Flow Module for more details on these formulations. Only one type of friction can be selected per grid. The user can either define one (uniform) friction value for the whole grid, or select a .ASC file containing friction values for all 2D grid elements as a way to model distributed friction. This .ASC file should have exactly the same format (including number of columns and rows) as the height definition file. It is possible, however, to have no-data values in 2D grid cells in the height .ASC file where there ARE values in the friction .ASC file. The other way around is not possible. Note that in the user interface, the number on the horizontal axis have been replaced by letters, ranging from 'A' to 'XX'.

The next tab, 'grid cell friction', displays a table with all friction values. This tab is only available when a distributed friction file has been selected first. It is not possible to modify any of the values seen here. If the user want to change any of these values, he/she will have to edit the friction .ASC file directory (outside SOBEK).

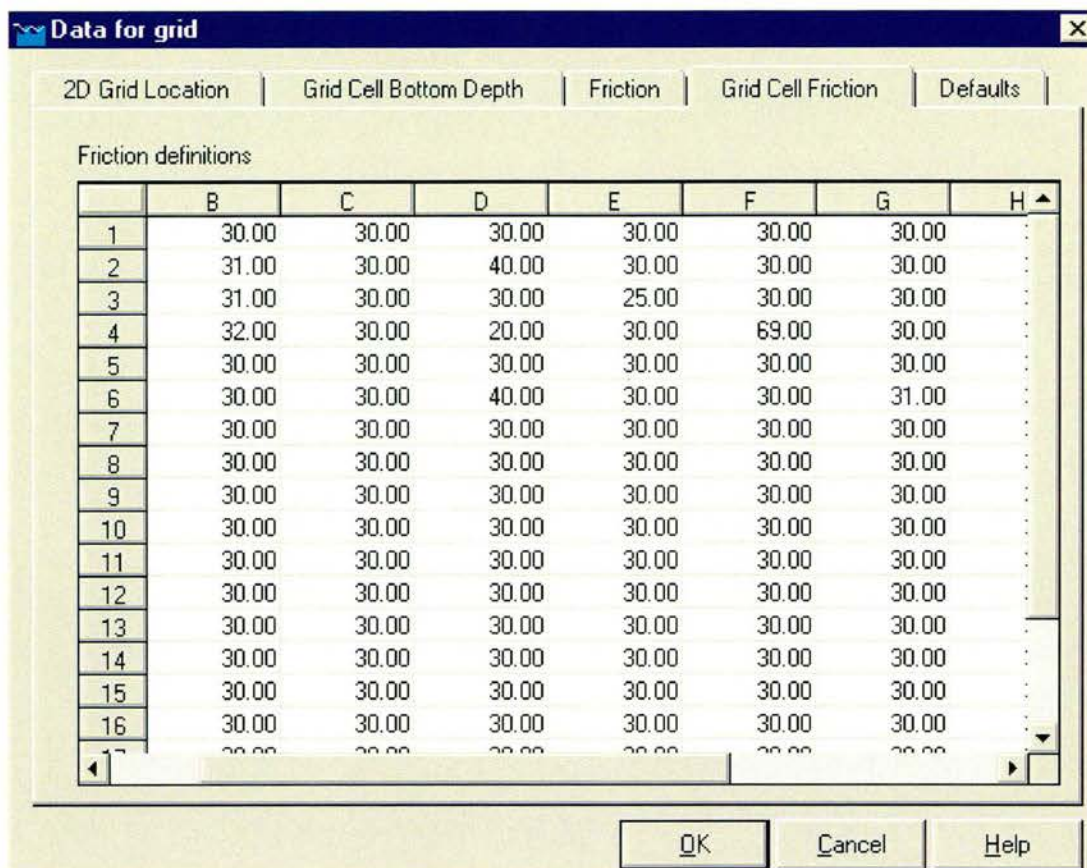


Figure 21b Grid data window - grid cell friction tab

The last tab, 'defaults', gives the user the possibility to save or load certain grid-related parameters as default. For example, if you save the 'friction' as default (only with uniform friction!), the next grid you define will have the same friction as the current grid.

2D - boundary node



After selecting any 2D-boundary node in the edit model data mode and pressing <edit>, the data edit window for this node type pops up. The first tab 'location' displays the (non-editable) id and location of node.

Under the second tab 'boundary condition', the user can specify the boundary condition. See Figure 3.22. There is a choice between a Water level boundary condition or a Flow boundary condition. Both of these can be given either a constant value or a value changing in time.

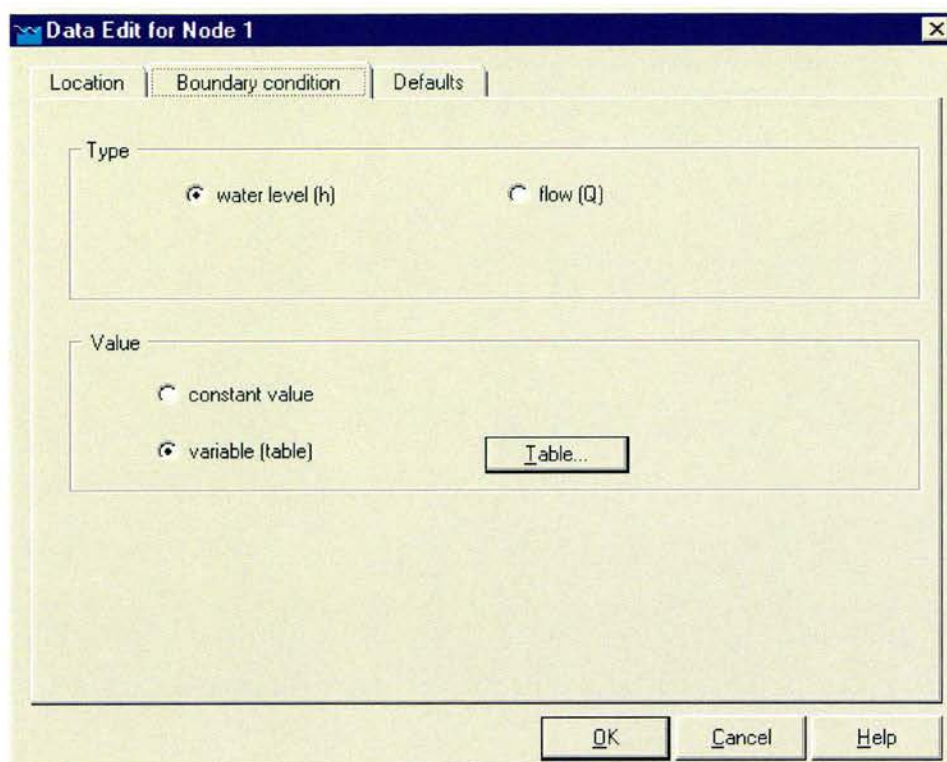


Figure 3.22 2D-boundary edit window - boundary condition tab

A variable waterlevel or flow in time can be entered by pressing <table..> and filling in the table, as shown in the example in Figure 3.23:

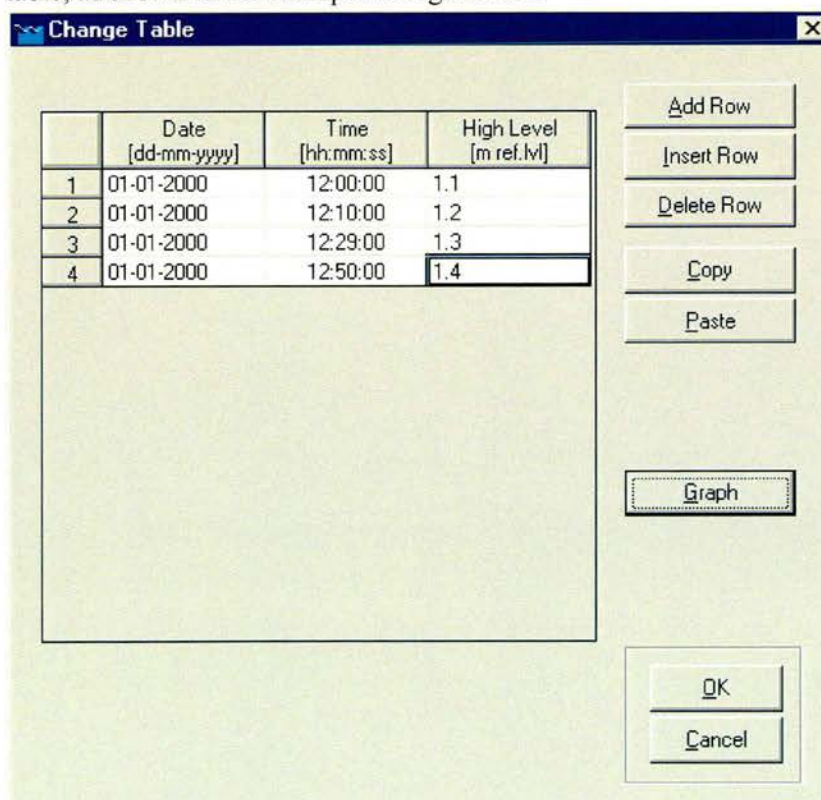


Figure 3.23 Change table window

Please note that the water level is defined relative to the reference level (positive above the reference level), and that it is not possible to enter a water depth as a boundary condition.

Also note that a positive value for the flow means input into the system, while a negative value means output.

2D - history node

44 2D - History



This type of node does not have any active parameters connected with it. It is purely a means for the user to select certain 2D grid cells where exact output is required.

2D - Breaking Dam node

45 2D - Breaking Dam



This node type gives the user the possibility to define the **decrease** in terrain level of a specific 2D grid cell as a function of time. An example of the data edit window with the table containing the levels is given in Figure 3.24:

	Date [dd-mm-yyyy]	Time [hh:mm]	High Level [m]
1	01-01-1996	00:00	0
2	01-01-1996	01:55	0
3	01-01-1996	01:56	3
4	01-01-1996	02:00	3

Figure 3.24 Model data for the 2D breaking dam node

Although not necessary, it is advisable to let the table cover the whole time interval of the simulation. Just as for all tables used in SOBEK, the program uses linear interpolation to determine the terrain level in between times specified in the table.

Note: the table contains **changes in the initial terrain level**, which are defined as **positive** in downward direction!

2D - Boundary Corner node / 2D - Line boundary

46 2D - Boundary Corner



The properties for the 2D-line boundary are exactly the same as for the 2D-boundary node, see Figure 3.22 and Figure 3.23. The user can define either a waterlevel or a flow. If a flow boundary is selected, this flow is attributed to every one 2D grid cell beneath the line boundary; so if you specify 500 m³/s for a line that covers 5 grid cells, the total input into the system will be 2500 m³/s.

2D - Initial Water Level Point node

47 2D - Initial Water Level Point



The only variable that needs to be specified for this node type is the initial water level, relative to the reference level. Note that a positive value means above the reference level, as opposite to terrain levels, which are defined as positive in downward direction.

Flow - Dam break branch

A dam break is simulated in two phases. At a certain moment, first, the gap crest level is going down with a constant gap width. When a certain maximum depth of the gap is reached, second, the width of the gap starts increasing. Figure 3.25 shows the different stages defined during the growth of the breach:

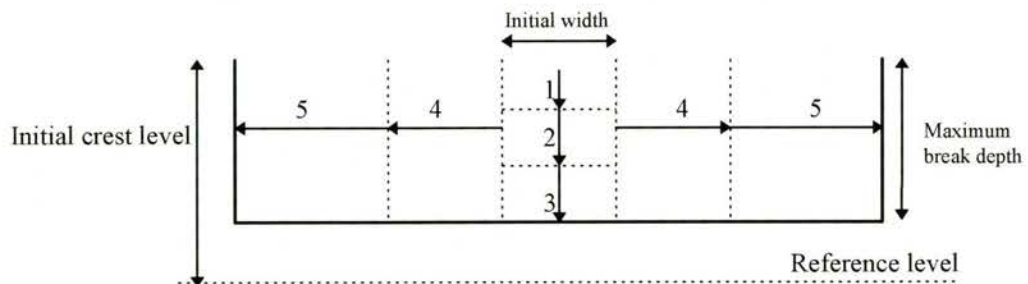


Figure 3.25 Breach growth

Figure 3.26 shows a typical chart of the gap area in time. Normally the first phase behaves as a linear function, while the second phase has a logarithmic course. The discharge through the dam break is calculated using the standard weir-formulas. Please refer to the technical reference manual of SOBEK Channel Flow for more details about the weir-formula.

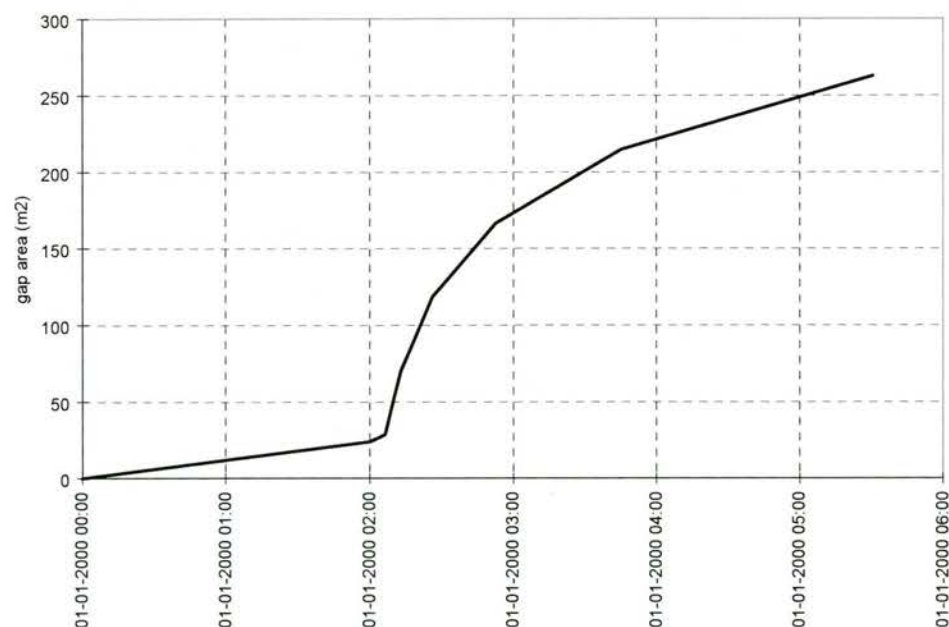


Figure 3.26 t-A-chart of gap of dam break

Figure 3.27 shows the data the user will need to supply for this type of reach. All data are related to the dam break concept explained before. Please refer to Figure 3.25 for an explanation of the initial width, maximum break depth and initial crest level.

Figure 3.27 Model data window for the dam break branch

The user can specify the time-Area table him/herself or choose to let SOBEK generate it, depending on a few parameters. These parameters are shown in Figure 3.27. The **period to reach maximum depth** refers to the first stage of the breach (linear), while the **end condition** (either **maximum width** or **period to reach max. width**) refers to the end of the second (logarithmic) stage. Please bear in mind that both periods start at **time start**, although they end at a different time.

After all the data has been entered the table can be generated with the **generate table** option. The table can then be viewed (and edited) in the **controller** tab.

In Appendix B the background principles of the calculation of the area are explained.

Some concluding remarks:

- The default maximum breach width when using sand equation is 200 m and for clay 75 m. If necessary, these values can be changed by editing the ini file `"sob_lite\programs\flow\SBKEDIT.FNM"` manually before starting SOBEK. It contains two keywords called "DamBreak1DMaxWidthSand" and "DamBreak1DMaxWidthClay" with values which control the maximum allowed widths. Keep in mind, however, that the formulas used to calculate the A-t table do not give reliable results for widths above these default maximum values!

- The **period to reach max. width** should be at least a 100 seconds longer than the **period to reach maximum depth**

3.3 Task block 3: 'results in maps'

3.3.1 General

The 'results in maps' task is the most important of the three task blocks when viewing and analysing results from a simulation. It is usually the first place to go to if you want a quick scan of the results, because the Netter environment provides you with many tools to help you. Once again, the user is expected to have a working knowledge of SOBEK and Netter before reading this manual. This paragraph focuses on explaining the additions to the Netter environment concerning the viewing and exporting of 2D data. It starts with explaining the use of the *active legend*, which is a great help to quickly understand exactly what results you are viewing and to turn data layers on and off. After that, some more information will be given concerning the three output formats: incremental (only for use within Netter) and GIS (for output to GIS applications). The third output format, History stations (for use in Netter or as output to for example EXCEL), is not explained any further in this document, as it is a standard SOBEK output format which is already explained in the regular SOBEK documentation.

3.3.2 Viewing model and output data in Netter

Active Legend

The *active legend* is one of the major additions to this new version of SOBEK/ Delft-1D2D. It can be switched on in Netter in both the *schematisation* task and the *results in maps* task, by selecting **options => active legend**. It is a great help with identifying and selecting output data.

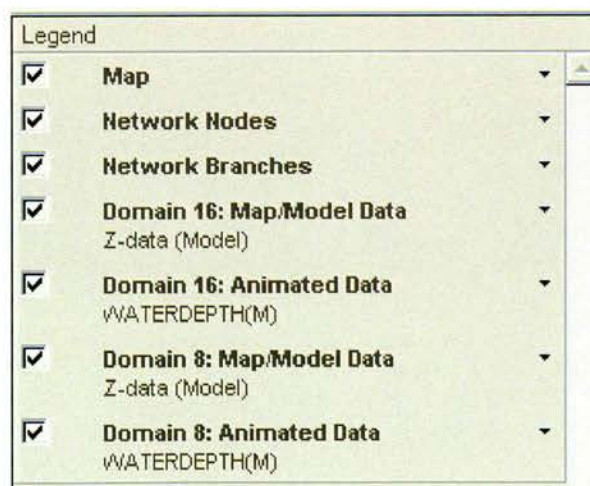


Figure 3.28 Active legend

Once the active legend has been switched on, the model and output data become visible as groups in the legend window (Figure 3.28). All groups can be turned on or off using the check-boxes. Selecting a group in the legend window activates a shortcut to the properties/settings of the group. The triangle-like symbol on the right hand side of the group names turn the group-legends on and off.

Depending on the schematisation and the type of currently selected output data (under **file** => **open data**), one or more of the following groups become available:

- map (model data)
The currently loaded map-layer.
- Network nodes (model data)
All nodes, both 1D and 2D, in the schematisation
- Network branches (model data)
All branches in the schematisation
- Map/model data for every grid (called 'domain')
This data layer can contain either model data OR output data:
 - z-data (model data): terrain levels with respect to reference level. Positive means downward. The topography .asc files are examples of the z-data.
 - depths/levels/velocities (output 'map' data): output results for a selected timestep. It is not possible to 'scroll' through the simulation time.
- Animated (output) data for every grid
This layer can contain any one of the following output data:
 - Depths
 - Levels: as opposed to the z-data, the levels are always defined as positive in Upward direction!
 - Velocities. Should always be positive.
- Network data
This layer can contain the results of a selected 1D-output option (i.e. *waterlevels at nodes*) OR the 2D results at history stations.

3.3.3 Incremental output files

Incremental files are generated by SOBEK mainly as a means to visualise animated results in Netter. However, it is also possible to generate an .ASC output file of the current timestep and for a selected grid via the **export** option (**file** => **Export** => **D2grid data** => **Animated data**). Remember that the incremental files contain classes instead of actual values, so the .ASC file generated as output will contain classes as well. In most cases it will be more useful to create .asc files from either a MAP file or use a GIS .asc file directly.

3.3.4 GIS/ MAP output files

These files are created mainly for the purpose of exporting output data to a GIS-based system such as ARCVIEW.

The GIS files are the .asc outputfiles that the user selected in the *settings* task. A distinction is made between 'normal' asc files and 'special' asc files. One 'Normal' asc file is generated automatically for every (in *Settings*) selected GIS output timestep and variable. One 'Special' asc file is generated for every special variable, but only one for the whole simulation period! The exception to this rule are the files for the dD/dt variable, which are generated every GIS output timestep. After the simulation has completed and the case has been saved, these files are written to the <casename>-directory. Remember that these files contain the actual values and not the classes. Each file represents the actual values for a particular variable and for a particular time.

The following table summarizes the available special .asc output files:

parameter	symbolisation	unit	filename	explanation
maximum water depth	d_max	m	dmNMAXDX.ASC/ dNNMAXDX.ASC	in each cell the maximum water depth of all calculated timesteps is written
time of maximum water depth	t_dmax	hr	dmNTMAXD.ASC/ dNNTMAXD.ASC	in each cell is written after what period this maximum water depth has been reached
maximum water level	h_max	m ref. level	dmNMAXHX.ASC/ dNNMAXHX.ASC	in each cell the maximum water level of all calculated timesteps is written
time of maximum water level	t_hmax	hr	dmNTMAXH.ASC/ dNNTMAXH.ASC	in each cell is written after what period this maximum water level has been reached
maximum velocity	c_max	m/s	dmNMAXCX.ASC/ dNNMAXCX.ASC	in each cell the maximum velocity (absolute value) of all calculated timesteps is written
time of maximum velocity	t_cmax	hr	dmNTMAXC.ASC/ dNNTMAXC.ASC	in each cell is written after what period this maximum velocity (absolute value) has been reached
time of wetting	t_wet	hr	dmNTWTXX.ASC/ dNNTWTXX.ASC	in each cell the <i>first</i> timestep is written at which the cell starts to become wet. If the cell doesn't become wet it remains '-999'
time of wetting per class ('permanentie')	t_wet/class	hr	dmNWTCIL.ASC/ dNNTWTCIL.ASC	for each incremental class in each cell the period is written during which this class is exceeded. If the class is never exceeded it remains '-999'
time of drying	t_dry	hr	dmNTEMXX.ASC/ dNNTTEMXX.ASC	in each cell is the <i>last</i> timestep written at which the cell starts to become dry again after wetting. If the cell doesn't become dry after wetting it remains '-999'
time of drying per class ('permanentie')	t_dry/class	hr	dmNEMCIL.ASC/ dNNTNEMCIL.ASC	for each incremental class in each cell the period is written during which the water depth is lower than this class. If the water depth is always higher than a class, it remains '-999'
rate of change of water depth	dd/dt	m/s	dmNRSSSS.ASC/ dNNRSSSS.ASC	in each cell the difference in water depth is written between two GIS output files divided by the GIS output options timestep Δt

Notice that in the files, where 'hr' is the unit, $t = 0$ is at the start of the simulation.

Example for the 'rate of change of water depth': if you run 10 hours of simulation period, and you choose in 'GIS output options' $\Delta t = 02:00:00$ hh:mm:ss, than this will result in 5 maps.

Filename structure

Most of the .asc filenames generated have the following structure:

dmNVSSSS.asc/dNNVSSSS.asc

General notations used for ASC and MAP files are :

'dm' or 'd' => an abbreviated form of 'domain' (another name for 'grid'), dm is used when domain number is between 1 and 9, and d is used for domain numbers between 10 and 99.

'N' => domain number, from 1-9. This is an internal number, which is not the same as the domain ID.

'NN' => If domain number is from 10 onwards till 99, then the name of the .asc file will begin with 'dNN' i.e for say domain 19 it will be d19VSSSS.ASC

To check which domain number is related to which domain ID, please refer to file FLSGIS.HLF file in your case directory.

'V' => type of variable:

'd' - depth

'h' - level

'c' - velocity

'u' - horizontal velocity

'v' - vertical velocity

'r' - $\Delta d/\Delta t$

'SSSS' => represents chosen output timestep number (not the timestep chosen for the calculation!). In other words, if the timestep selected is 15 minutes, then at start of simulation, SSSS = 0000, at 15 minutes after the start of simulation, SSSS=0001 and so on.

'XXXX' or 'XX' or 'X' => represents 0 or 1 placed to make the filename 8 character long.

'II' => represents the class interval number as given in the Incremental file for a selected parameter.

Similar to the ASC file a MAP file is also created with the name

dmNVXXXX.map/dNNVXXXX.map

The main differences between the ASC and MAP file are that ,

MAP file is a binary file as opposed to ASC file

MAP file contains actual values for all timestep, while one ASC file contains actual values for a one timestep. Hence no matter how many timesteps are defined by the user, there will be only one map file created.

There are a number of other .ASC file created for variables like maximum water depth/ maximum water level/ time at which maximum depth occurs/ time at which maximum water level occurs etc.. These files can be identified from their filenames:

The filenames for such variables are:

Maximum water depth

dmNMAXDX.ASC/dNNMAXDX.ASC
dmNTMAXD.ASC/dNNTMAXD.ASC

The maximum water depth value for the whole simulation run is written in dmNMAXDX.ASC/ dNNMAXDX.ASC while the corresponding timestep when the maximum water depth occurs is written in dmNTMAXD.ASC/ dNNTMAXD.ASC. These files are generated when the option of water depth is checked on in Settings.

Rate of change of water depth

dmNRSSSS.ASC/dNNRSSSS.ASC

The rate of change of depth per selected output timestep is written in dmNRSSSS.ASC/dNNRSSSS.ASC. These files are generated when the option of water depth is checked on in Settings. The timestep at which these files are written is same as the selected timestep for incremental file and thus irrespective of what the output timestep is for generation of GIS/MAP files in settings.

Maximum velocity

dmNMAXCX.ASC/dNNMAXCX.ASC
dmNTMAXC.ASC/dNNTMAXC.ASC

The maximum velocity value for the whole simulation run is written in dmNMAXCX.ASC/ dNNMAXCX.ASC while the corresponding timestep when the maximum velocity occurs is written in dmNTMAXC.ASC/ dNNTMAXC.ASC. These files are generated when the option of velocity is checked on in Settings.

Maximum water level

dmNMAXHX.ASC/dNNMAXHX.ASC
dmNTMAXH.ASC/dNNTMAXH.ASC

The maximum water level value for the whole simulation run is written in dmNMAXHX.ASC/ dNNMAXHX.ASC while the corresponding timestep when the maximum water level occurs is written in dmNTMAXH.ASC/ dNNTMAXH.ASC. These files are generated when the option of water level is checked on in Settings.

Time of wetting/drying

dmNTWTXX.ASC/dNNTWTXX.ASC
dmNTEMXX.ASC/dNNTEMXX.ASC

These two files are special output file. The values in the file dmNTWTXX.ASC/ dNNTWTXX.ASC gives the maximum time span the grid cell remains flooded i.e wetted and dmNTEMXX.ASC/ dNNTEMXX.ASC gives the maximum time span the grid cell remains dry i.e empty. These files are generated when the option of water depth is checked on in Settings.

Time of wetting/drying per class

dmNWTCIL.ASC/dNNWTCIL.ASC
dmNEMCIL.ASC/dNNEMCIL.ASC

These two files are special output files. The values in the file dmNWTCIL.ASC/ dNNWTCIL.ASC gives the timestep when a water depth in a grid cell is high enough to be in the particular class. For examples: if there are three classes for depth values i.e. less than 0.1 m, greater than or equal to 0.1 m but less than 1.0 m, and greater than or equal to 1.0 m. So in total three files will be generated for each domain i.e dmNWTC01.ASC/ dmNWTC02.ASC/ dmNWTC03.ASC. When a grid is just flooded but the water depth is still less then 0.1 m then the timestep at which the grid cell is flooded is written in dmNWTC01.ASC, when the water depth in grid cell becomes just higher or equal to 0.1 m but less than 1.0 m that timestep is written in dmNWTC02.ASC, and when the water depth becomes just higher or equal to 1.0m that timestep is written in dmNWTC03.ASC.

Similarly the values in the file dmNEMCIL.ASC/ dNNEMCIL.ASC the timestep when a water depth in a grid cell is receding and is low enough to be the particular class.

These files are generated when the option of water depth is checked on in Settings.

3.3.5 Output for 1D dam break reach

The output for the 1D dam break reach can be found under 'open data' - 'flow module: levels and flows' - 'structures'. Once this option has been selected, the dam break reach can be selected and the following output can be viewed:

- the discharge through the structure
- the upstream water level
- the downstream water level
- the breach area in time (A-t). At the moment, this variable is accidentally called 'Cr. Lev. / Op. Hgth.' in SOBEK!

4 Frequently Asked Questions

4.1 Modelling-related questions

4.1.1 When do I choose dikes in the 1D profiles?

In Delft 1D2D it is possible to choose the option of a dike within the definition of the 1D profile. The switch applies for the whole model, so all 1D profiles use dikes or not. This option is useful in cases where it is not desirable to model the dikes in the 2D part, for example if the riverprofiles are too detailed to schematize with the 2D grid. The 1D profile definition is valid until the crest of the dike. From there the grid should contain the data of the floodplains along the dikes. If dikes are chosen, which tower above the ground level, the 1D line elements function as a barrier for water coming from the 1D part as well as from the 2D part.

In cases where the 1D profiles have no important dikes or levees and the local exact connection of the 1D profile with the ground level is not known, it is useful to use “no dikes” in the 1D profiles. In that case the profile can be defined 1D independent of the grid level and the model uses this 1D information until it intersects the bottom level of the 2D grid. In this case the dike above the 2D grid level is transparent, for water coming from the 1D part as well as the 2D part.

One thing to bear in mind when using the dike option, is that **the dike goes all around the 1d channel!** With the dike switch **on** in the example of Figure 4.1, no water will flow out into the 2D grid from the end connection node of the dummy branch, until the dummy branch has filled up to the level of the dike. So the channel at the end connection node is 'closed' until the height of the side profiles!

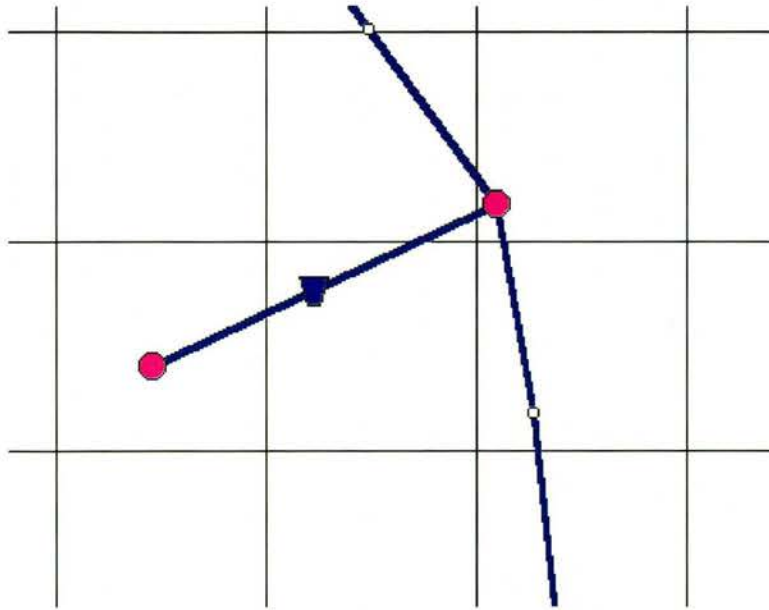


Figure 4.1 Example of 1d dike

One way to solve this particular problem is shown in Figure 4.2:

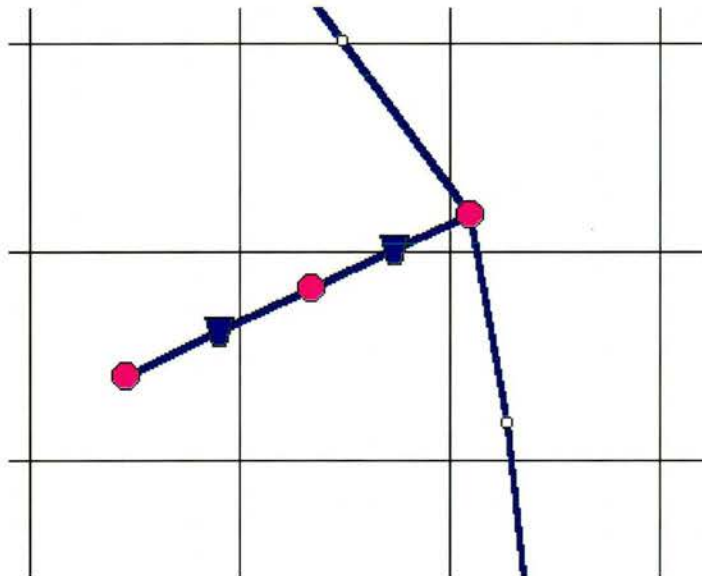


Figure 4.2 Example of lowered dike at end

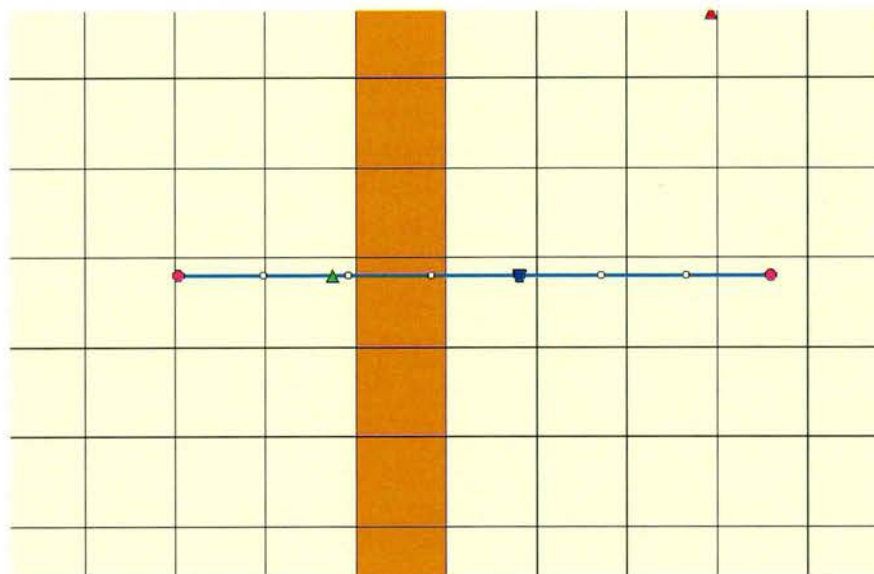
The dummy branch has been split into two separate branches. The downstream branch has a profile that is much lower than the original dummy branch profile. This way, water will be able to leave the 1D channel without a barrier at the end of it via the last connection node.

4.1.2 How do I visualize the bottomlevels of the 2D grid together with the bottom level of the profiles?

In Netter (Schematisation, Edit model) you can add labels showing the bottom levels of the profiles of your 1D network. To do this use: Options, Network Data, Nodes: Other: choose 'Level'.

4.1.3 How do I model culverts

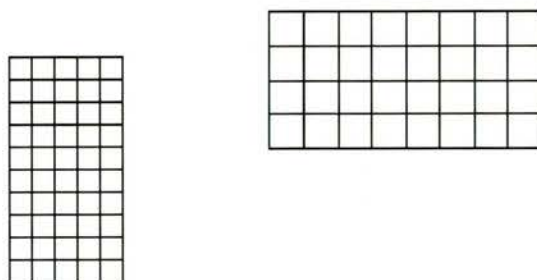
If there are culverts under a road or dike it can be schematized by using a short unconnected 1D branch with an orifice and a rectangular closed profile crossing the 2D line element. The orifice takes into account the entrance losses. The closed-conduit's roughness covers the remaining losses.



4.1.4 How do I model multiple grids?

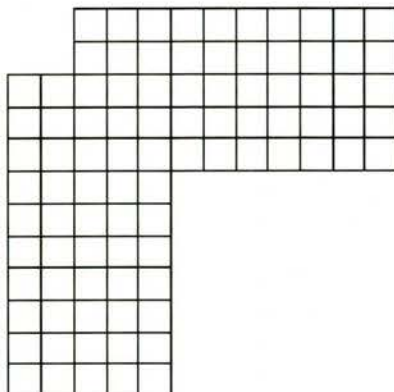
Using Delft-1D2D, the user can specify multiple grids within one case. There are a number of configurations possible:

- Multiple, separate grids



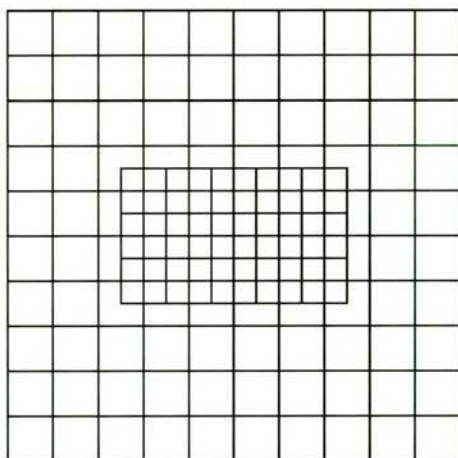
The grids do not have any overlap, so no water will flow from one grid to the other. The user can define as many grids as necessary.

- Multiple, overlapping grids



The grids have an overlap. Only two levels of overlap are possible. There can be as many 'single level' overlaps as necessary. Water will flow from one grid to the other. Remember that the outside grid cells are removed from both grids, so in the example shown above both grids have actually only ONE cell in common!

- Multiple, nested grids



One (or more) grid is completely situated within another grid. The nested grid is called the *child* grid, the other grid the *parent* grid. Only one level difference is possible, so you cannot define a child grid within another child grid. Water will flow from one grid to the other. All grid elements of the child grid should fit **exactly** within the grid elements of the parent grid, except for the outer layer of cells which is removed from the schematisation. For example, if you look closely at the position of the child grid inside the parent grid, you will notice that the child grid fits exactly inside **4x2** grid cells of the parent grid, after the outer grid cells of both grids have been removed by SOBEK.

Important note: due to an error, it is advised **NOT** to use the automatic 'align to parent' and 'fix parent' options available in the edit network mode!!!

4.1.5 How do I model a dike-breach using the 1D channel flow module?

Modelling a dike-breach using the 1D module has many advantages compared to using only the 2D overland flow module. First of all, the river can be modelled very accurately in 1D using the exact dimensions of the profiles.

This usually means that the 2D grid cells can be larger compared to when the river is modelled in 2D, which in turn means smaller calculation times, besides more accurate results. Secondly, a special 1D dike-break node has been developed, which gives the user a very flexible way to model the horizontal growth of the breach. Thirdly, there are numerous other types of nodes available within the 1D channel flow module (like the weir, orifice, bridge and culvert nodes), which make the life of the modeller even more easy.

When the modeller has decided to use the 1D channel flow module for the dike-breach, there are many different ways to model the phenomenon. One example is described below.

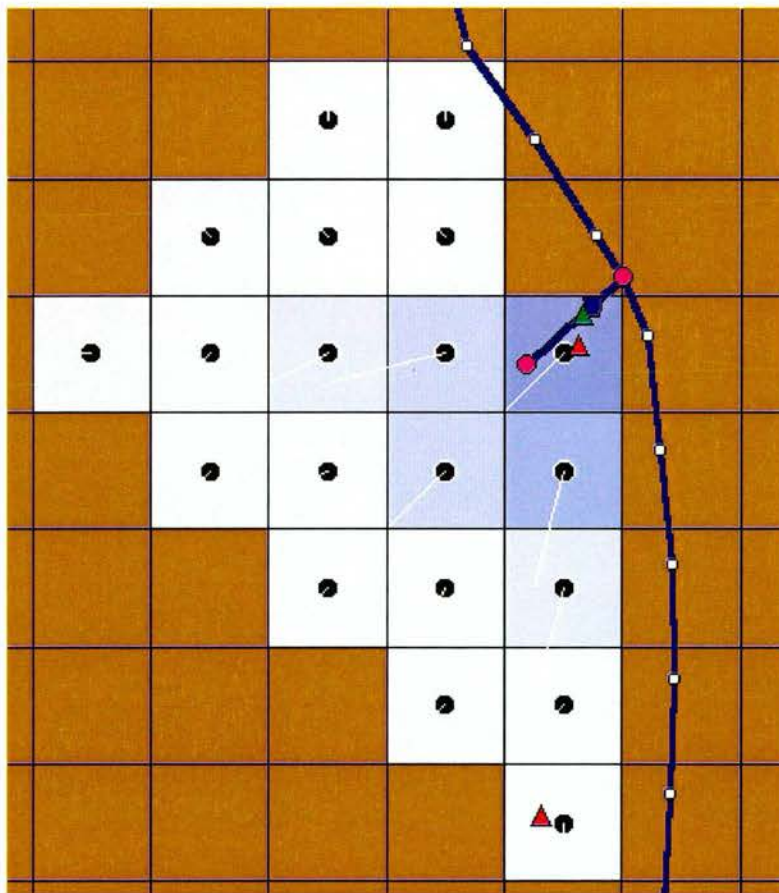


Figure 4.3 1D dike breach using a 1D dummy branch

In this example, a 1D dummy branch is added to the 1D channel system at the location of the dike breach. Its purpose is to prevent water from entering the 2D grid on the other side of the river, which would happen in the case of modelling the dike breach in the main river.

This can be explained as follows:

The (white) calculation-point to the south of the (red) connection-node in the main river is connected to the underlying 2D grid cell. Now, suppose the dike breach starts at this point. Water will start to enter the 2D system via this particular grid cell, and it will flow into the surrounding cells on BOTH sides of the river! Usually, a dike breach will only occur on one side of the river / channel, so that's why a dummy branch is used. Because now, the water will flow via the (red) connection node from the dummy branch into the grid cell next to the river.

Because the embankments of the main 1D river are higher than the 2D terrain level, the water will stay on one side, and not flow to the other side through the river. This is clearly visible in Figure 4.3.

The actual breach itself is modelled in this case by using a standard 1D weir node with a time controller, which lowers the crest level in time. Because the profile in the dummy branch is lower than the profile of the main river, water will leave the river via the end of the dummy branch once the crest level of the weir has lowered.

Using the 1D dam-break instead of the weir node has the advantage that the width of the breach can be defined in time as well.

4.1.6 How to interpret the 'Simulation messages'

It is advisable to always check the 'simulation messages' after a simulation, even if it has run successfully. The computational core of SOBEK creates a log file called 'SOBEK.log', which contains messages and warnings concerning the simulation. In some cases, SOBEK will change properties of certain objects in the schematisation in order to be able to run the simulation properly. Usually these changes are necessary because of mistakes in the schematisation. SOBEK logs every change made.

One example of a message that might occur is the '**negdpt...**' message. This message appears whenever a negative water depth occurs in a system after solving the equations. In such a case the program itself tries to lower the internal computational timestep so that the negative water depth does not occur. It may happen that after a few tries the negative water depth still occurs. In this case, the program makes the water depth just equal to zero and goes further on with the computation in such a way the total water mass balance is achieved. This will cause the computation run to slow down considerably. When you get this message in the SOBEK.log file, it is a good idea to check your schematisation, as there might be something wrong with it. You should check the following point:

- There should not be more than one 1D network points (connection nodes / calculation points) lying within a particular 2D grid cell.

4.2 GIS-related questions

4.2.1 How do I import a *.shp-file as (part of) the network?

In Netter it is possible to import *.shp-file: File, Import, type the name of the file together with the extension 'shp', for example "provlyn.shp"). Importing shape files is only possible in case of polyline elements, not with point or polygon elements.

4.2.2 How do I edit the 2D grid?

Small adaptations to bottom levels of the 2D grid can be done within Delft 1D2D. In Netter (Schematisation, Edit model) visualize the grid's Z-data, point with the mouse to the cells which you want to edit. Right below your screen two numbers appear: the column-number and the row-number. Determine which cells should be edited. In Edit, Model data, Grid, Edit, Grid cell Bottom level you can search in the table the particular cell and change its value. For extended adaptations to the 2D grid a GIS-package should be used.

4.2.3 How do I model roads and railway dikes?

These infrastructural elements should be part of the 2D grid. To apply these elements to the grid a GIS package is advisable, but you can also change the cell heights by hand in the .asc file.

4.2.4 How do I delete a grid?

Netter treats a grid as if it were a node. So you can delete a grid as you would delete a node, by selecting **delete node** from the *edit network* menu. This is only possible after first having made the grid nodes visible. Select **Options** => **Network options** => **Nodes** => select **2D-grid** and make it visible (see Figure 4.4).

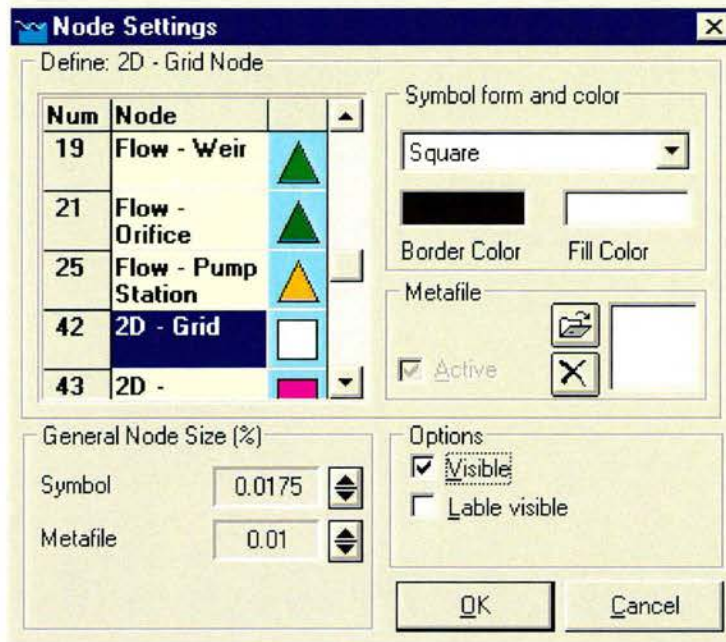


Figure 4.4 Making the 2D grid nodes visible

4.3 Scale-related questions

4.3.1 Which gridsize should I use in the schematisation?

The idea behind Delft - 1D2D is to model all details in the 1D part and to keep the 2D part rather coarse. This will reduce the necessary calculation time. The calculation speed is for a great part determined by the number of 2D grid cells. If your simulation times are too long, consider to use a coarser grid.

The 1D profiles should not be wider than the grid size. If they are, remember that the winter bed of broad rivers with vast floodplains should be modelled in the 2D part. The summer bed can be modelled in 1D. For polder channels or relative narrow channels/canals (compared to the 2D grid) the 1D part can should be used.

4.3.2 What is the maximum possible number of grid elements?

A number of tests have been conducted to find out what the maximum number of 2D grid elements is that can be used for a simulation. The first test, with a grid of 170,000 nodes didn't cause any problems. The average calculation time for every one timestep (of one minute) was 16.5 seconds. The second test, with 360 000 nodes (600 x 600 grid elements) ran fine, except that the simulation time was rather long (25 minutes per timestep of 1 minute). The last test, using a grid of 490,000 nodes (700x700 elements) caused problems, especially in the postprocessing (Netter). It is therefore recommended to try to keep the total grid size limited to around 360,000 nodes.

Note: all tests were carried out on a Windows95 computer with a PII 450 mhz. processor and 128 mb. of internal memory.

4.4 Installation-related questions

- What to do when wanting to copy a project to a drive with another drive letter (i.e. drive c instead of d)?
 - keep the directory on the old drive under which the ascii-files are stored
 - to keep the ascii-files together with the projects, copy the relevant ascii-files to the fixed directory c:\sob_lite*.lit\fixed, in the project directory
 - open the case in the c-version. Delete the grid, and import it from c:\sob_lite*.lit\fixed.
 - make sure to always check the grid data after importing a grid. You will also have to redefine the friction.

5 Tutorial Delft-1D2D (SOBEK-Channel Flow and Overland Flow modules)

The Overland Flow module included in SOBEK-Lowland is meant to carry out two dimensional flooding simulations as its predecessor Delft-FLS (Flooding System). Its advantage in relation to Delft-FLS is the fact that it can be automatically combined with the 1D features of SOBEK. While using this new system, you will often encounter the term “FLS” used in this case as a synonym of a 2D feature.

The combination of the existing Channel Flow module of SOBEK and the whole SOBEK UI with this new Overland Flow Module has also been called “Delft-1D2D”.

The basic principles of working with the Overland Flow Module of SOBEK are explained step by step in this tutorial. We assume you have already gained some insight in the principles and use of the SOBEK CF-RR modules (Channel Flow - Rainfall Runoff), explained in the previous tutorial.

In this tutorial, you will be guided to set-up a simple 1D network and to combine it with a simple 2D grid. This tutorial will only show a limited number of the large amount of options. It will teach the basic principles of working with SOBEK-CF-OF (also called Delft-1D2D system) and give you enough experience to continue on your own. Little experience on working with WINDOWS is required.

The tutorial contains:

- combined settings;
- setting up a simple network;
- adding a 2D grid;
- running a simulation;
- visualisation of results.

The tutorial does not explain all options in all appearing windows. Once you get the hang-and-feel of the model, you may wish to browse through the options not dealt with in the tutorial.

5.1 Setting up of the combined system

Start a new project (for instance tut1D2D; the extension .lit will be added automatically) and, inside the project, start a new case.

At the **Import Network** menu, select “start from scratch”.

At the **Settings** menu you will be able to activate the two dimensional “Overland Flow” (OF) module. Notice that you cannot choose the Overland Flow module alone. It will always work in combination with the “Channel Flow” (CF) module (See Figure 5.1).

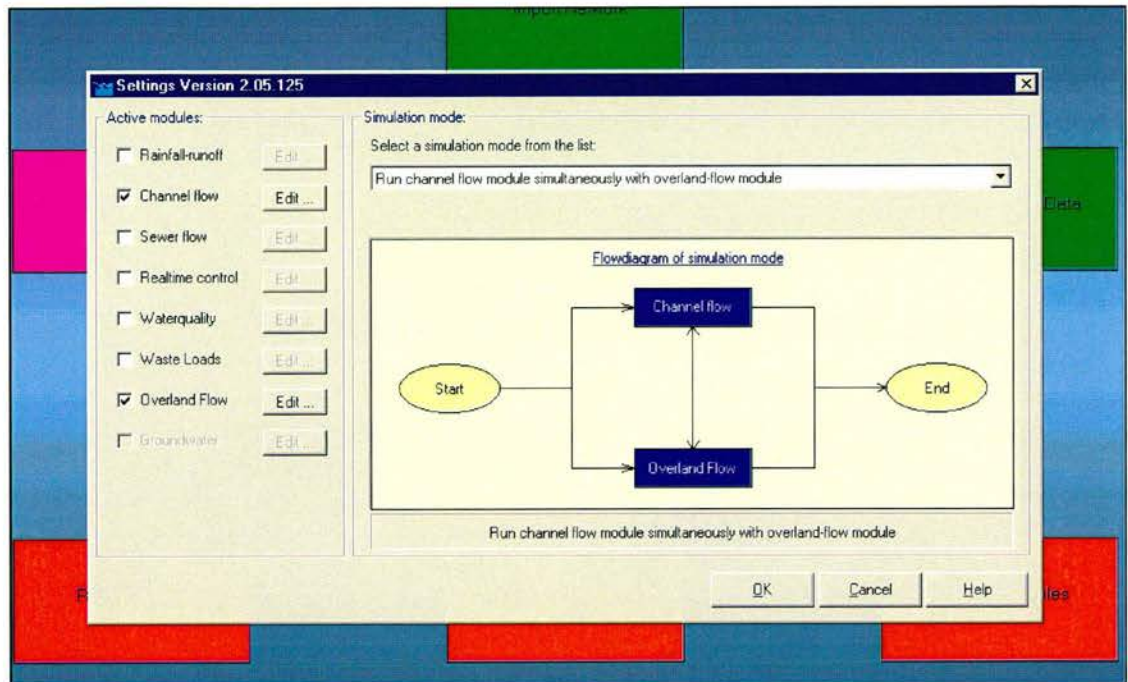


Figure 5.1 Settings, choice of the simulation mode

Note: When you want to simulate a pure 2D case without a 1D network, select both modules on the settings menu, and then make a “dummy” 1D branch. This dummy branch does not need to be connected to the 2D model. Give this dummy branch the necessary dimensions (profile and roughness) in order to be sure that flow will never occur within it. This dummy branch will not interfere with your case, but it will guarantee the good functioning of the system.

At the moment of releasing this version, no communication is still available among the CF-OF modules and other SOBEK modules such as Rainfall-runoff (RR), Sewer Flow (SF), Real time control (RTC), Water Quality (WQ), Waste Loads (WLM) or Groundwater.

The settings within the Channel Flow module have already been explained in the previous tutorial. Choose for “Time settings / Simulation period” the option “Simulation period defined as below” since this version does not yet allow the use of the Rainfall runoff and the Overland flow modules simultaneously.

The 2D Overland Flow settings are divided in three information sections:

- The time settings. They are fixed and have already been defined in the CF time settings
- The initial data. They can be defined in the schematisation of the 2D-grid (Schematisation / Edit network / 2D-Initial values)
- The 2D Output options

The 2D Output options allow the choice of the following output parameters:

- GIS: map and .asc output at each time step or at multiples of the time step for water depth, water level, velocity (total), velocity u-component, velocity v-component (map: the value of the chosen parameter is given at every grid cell at a certain time step).
- History: the desired parameter will be given as a time series at a selected history station (station predefined within model schematisation); the user can choose to write the results every time step or at multiples of it.
- Incremental: the results for the 2D cells are always saved in an incremental file. The program does not save the computed value of a variable at each time step, but it saves the information at the moment that the variable changes from one predefined class to another. For example, the computed water depths can be classified within classes: (< 0.02 m: dry), (> 0.02 to 0.10 m), (0.10 to 0.50), (0.50 to 1.00 m), (1.00 to 2.00), (> 2.00). While a cell is dry, no results will be written to the incremental results file. When the 0.02 m is reached, this change will be given to the output file. The same when the water depth passes to the next class, and so on. The user can choose which results will be stored in the incremental file and define the classes for each of them. The output can be specified at each time step or at multiples of it (See Figure 5.2).

Settings for Overland Flow module

Simulation settings | Initial data | **GIS Output options** | Incremental Output | History Output

Timestep output (same as Channel Flow)

Each timestep(s) This is similar to (dd:hh:mm:ss:)

Output Parameters

☒ water depth
☐ water level
☒ velocity
☒ velocity(u component)
☒ velocity(v component)

Classes for incremental file:

Parameter:
 Number of classes:

H <	-0.01
-0.01 < H <	0.1
0.1 < H <	0.2
0.2 < H <	0.3
0.3 < H <	0.4
0.4 < H <	0.5
0.5 < H <	0.6
0.6 < H <	0.7
0.7 < H <	0.8
0.8 < H <	0.9
0.9 < H <	1

OK Help

Figure 5.2 Output to the incremental file. Definition of classes

At the “**Meteorological Data**” Menu, no special data will be necessary, at least when working with boundary conditions given by the user. The input within this menu is required when working with the Rainfall-runoff module.

The steps to be followed at the “**Schematisation**” Menu / Edit model are described in the next sections.

5.2 Setting up the 1D network

As most of the features of SOBEK Channel Flow have been already explained, no special attention will be given to the steps to be followed while building a 1D network. Only the elements of the 1D network necessary to follow this tutorial are given here.

Note:

When starting a complete new model, the standard map from the Netherlands is provided by Sobek-Lowland as background. If we want to work with another map, some special arrangements should be made: While being at the box “Schematisation”, menu “Edit model / Check Flow Model / View Data Flow Model / OK”, go to the directory ... \Sob_lite\yourproject.lit\Cmtwork and edit the file ntrpluv.ini., Remove from the third line the text “edit network”. Save this file. Go back to Sobek CMT (Case Management Tool) \ Schematisation menu and enter into the Edit Model option. Netter will appear. Using the option File / Open you will be able to combine different maps or mpl (Maplink standard) layers to your network (ntw) or to .bna vector files (see manual Netter).

1D network used in this tutorial:

At the CMT-box “SETTINGS”, time settings, choose:

Time settings: start 1999/01/01; 00:00:00, end 5 hours later; time step = 1 minute.

Within the CMT-box “SCHEMATISATION”, Edit Model (Netter):

Make three branches connected to each other (see Figure 5.3). The last one (right hand side) must be 10000 m long (start co-ordinates left hand side: x 257170, y 585050; end co-ordinates right hand side: x 267170, y 585050).

Define calculation points at the branches. Specially at branch 3 define calculation points every 100 m.

The extremes of the first and third branches will be boundary nodes.

Define a tabulated cross-section profile with terrain level -5 m, width 100 m, and top level +3, width 100 m (horizontal channel, rectangular cross-section given in a table form). The same cross section will be used for the first two branches. Assign to the third branch a cross-section with a uniform width of 100 m, terrain level -5 m, top level -3. Define the friction using the Chezy coefficient $45 \text{ m}^{1/2}/\text{s}$. The initial value for the 1D branches will be a water level at -1 m.

Define at the upstream boundary condition (start branch 1, left side) a discharge boundary with a constant value of 350 m³/s. The downstream boundary (right extreme of branch 3) will be a closed boundary indicated by defining a constant zero discharge at this point.

Define a weir at branch 2 with the following characteristics: width = 60 m, crest level = + 2 m, discharge coefficient and lateral contraction coefficient equal to 1 (both positive and negative flow directions possible).

Define for this weir a time controller functioning according to the following table:

Date (dd-mm-yyyy)	Time (hh:mm:ss)	Crest level (m above reference)
01-01-1999	00:00:00	2
01-01-1999	00:30:00	2
01-01-1999	00:50:00	-3
01-01-1999	01:00:00	5

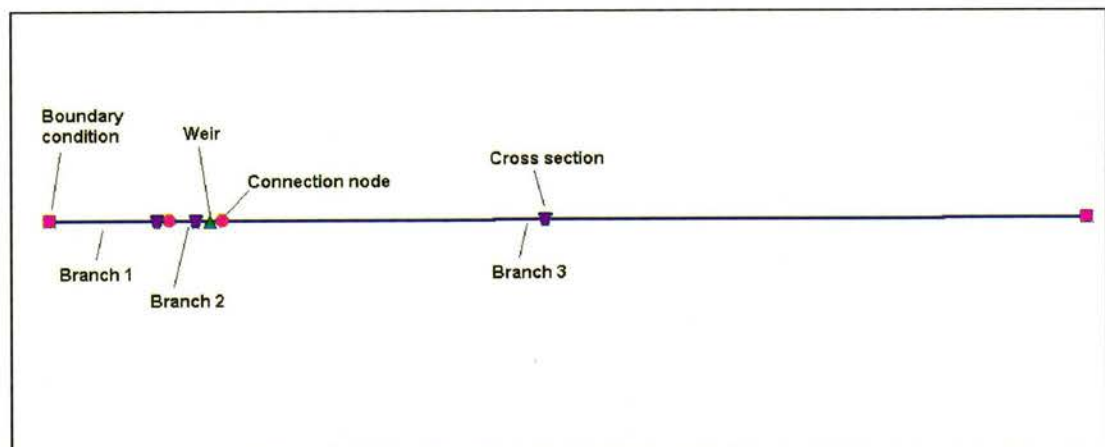


Figure 5.3 1D network formed by 3 branches

5.3 Setting up the 2D grid

5.3.1 2D grid

In order to add or to define a 2D grid (also called FLS-grid), enter at Netter into the Edit menu, Edit Network Submenu. Select from the list the 2D grid white square (Figure 5.4).

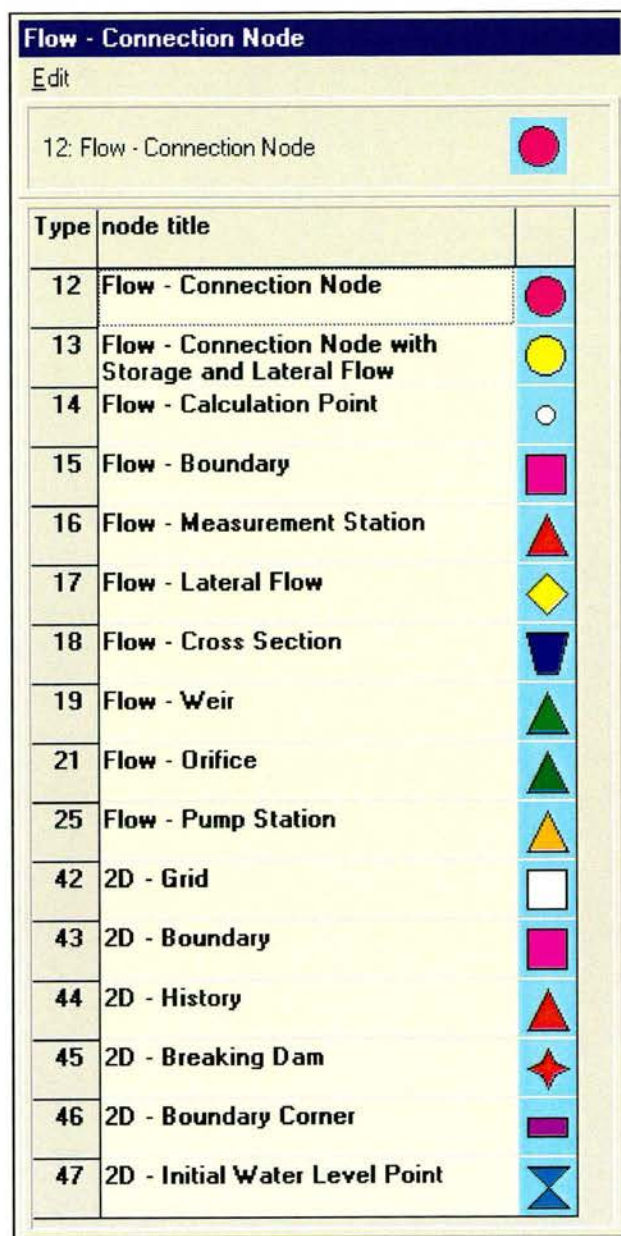


Figure 5.4 Edit / Edit Network / node types

Further on, you may proceed by entering into the Edit menu given at the top of the same window. The same can be done by pressing the right mouse button anywhere on your screen. While editing, you will be able, for example to change some settings (as automatic or manual name or ID), add or delete nodes, connect nodes, etc.

2D grid

Go to the Edit mode. There you will be able to:

- Define a 2D grid
- Import a 2D grid
- Edit a 2D grid (explained in 3.2.3).

Import a 2D grid

In this case the system is asking the user to give the name of the file containing the 2D grid information. This is an ASCII file with the extension .asc (for example gridtest.asc). This file has to be previously stored at the subdirectory ..\Sob_lite\projectname\fixed\yourfile.asc. It is identical to the file Arc-Info file required by the program Delft-FLS. An example is given below:

```

nCols      8
nRows      10
xllCorner   255000
yllCorner   574600
CellSize    100
nodata_value -999.000
-999.      -999.      -999.      -3      -3      -3      -3      -3
-999.      -999.      -3      -3      1      1      1      -3
-999.      -3      1      1      1      1      1      -3
-3      1      1      1      1      1      1      -3
-3      2      2      2      1      1      1      -999.
-3      2      2      2      1      1      1      -999.
-3      0      0      0      0      0      0      -999.
-3      0      0      0      0      0      0      -999.
-3      0      0      0      0      0      0      -999.
-3      -3      -3      -3      -3      -3      -3      -3

```

The file indicates the number of rows ($nRows = n$) and number of columns ($nCols = m$) of the grid, the geographic co-ordinates of the left-top corner, the cell size (square cell) and the value understood by the system as no-data value. After this the “depths” are given. Following the old Delft-FLS habit, mostly used in the 2D modelling world, the topography is given as depth measured from a certain reference level (mean sea level, NAP for The Netherlands). A positive depth value means in fact a negative level; negative values indicate grounds above the reference level. SOBEK-Overland Flow makes the internal transformation in order to work with the same level definition as used by SOBEK itself (positive level above reference).

Define a 2D grid

In this case, when you are asked to give a file name, type in a new file name in the directory ..\Sob_lite\<projectname>\fixed\..(ie.newname.asc). The new grid will be created. You will be asked to give the number of rows and columns, the co-ordinates, the grid size and the no-data value.

Figure 5.5 shows the first of the Grid Data sub-menu's. This information has been defined in the Edit Network option (manually or imported) and cannot be changed here.

The screenshot shows a software window titled "Data for grid" with a close button (X) in the top right corner. The window has five tabs: "2D Grid Location" (selected), "Grid Cell Bottom Depth", "Friction", "Grid Cell Friction", and "Defaults". The "2D Grid Location" tab contains two main sections: "Identification" and "Location".

Identification section:

- ID : 16
- Name :

Location section:

- Left (X) coordinaat : 177367.5625
- Left (Y) coordinaat : 480413.96875
- Corner Position : 0
- Number of Grid Points : 20
- Distance between Two Grid Points : 100
- in (X) Direction : 20
- in (Y) Direction : 20
- ARC Info File : \Sob_lite\DEMO.lit\FIXED\parent2.asc

At the bottom right of the window are three buttons: "OK", "Cancel", and "Help".

Figure 5.5 Model data / Grid data - 2D grid location

Figure 5.6, shows the layout of the 2D Grid Bottom Level screen. When importing a grid, the terrain levels will be automatically shown here; when defining the grid yourself, you will have to type (or copy/paste from spreadsheet) the terrain levels at this stage (here you can choose to use depths or levels !).

Gegevens voor grid

2D Grid Location | **2D Grid Bottom Level** | Friction | Friction | Defaults

☒ Bottom Levels in m wrt NAP ☐ Bottom Depths in m wrt NAP

	1	2	3	4	5	6	7
1	3.00	3.00	3.00	3.00	3.00	3.00	3
2	3.00	3.00	3.00	3.00	3.00	3.00	3
3	3.00	3.00	3.00	3.00	3.00	-3.00	-3
4	3.00	3.00	3.00	3.00	3.00	-3.00	-3
5	3.00	3.00	3.00	3.00	3.00	-3.00	-3
6	3.00	3.00	3.00	3.00	3.00	-3.00	-3
7	3.00	3.00	3.00	3.00	3.00	-3.00	-3
8	3.00	3.00	3.00	3.00	3.00	-3.00	-3
9	3.00	3.00	3.00	3.00	3.00	-3.00	-3
10	3.00	3.00	3.00	3.00	3.00	-3.00	-3
11	3.00	3.00	3.00	3.00	3.00	-3.00	-3
12	3.00	3.00	3.00	3.00	3.00	-3.00	-3
13	3.00	3.00	3.00	3.00	3.00	-3.00	-3
14	3.00	3.00	3.00	3.00	3.00	-3.00	-3
15	3.00	3.00	3.00	3.00	3.00	-3.00	-3
16	3.00	3.00	3.00	3.00	3.00	-3.00	-3
17	3.00	3.00	3.00	3.00	3.00	-3.00	-3

OK Cancel Help

Figure 5.6 Model data / 2D grid / 2D Grid Bottom level data

Further on, you may provide the bed friction data as a constant or varying in space using three different formulations (see Figure 5.7). When choosing a variable friction factor, a second friction window is activated. There you will be able to give the name of the friction file; this has the same format as the terrain levels file and has to be located at the same subdirectory (\Sob_lite\<projectname>\fixed\yourfriction.asc).

Remark!: at the moment, the file has to be generated beforehand.

Data for grid

2D Grid Location | Grid Cell Bottom Level | **Friction** | Grid Cell Friction | Defaults

Friction value (bottom)

Friction type

☒ Chezy

☐ Manning

☐ White Colebrook

☒ Constant value : 45

☐ Variable

Friction value (side wall)

☒ Constant value : 0

OK Cancel Help

Figure 5.7 Model Data / 2Dgrid / friction coefficients

Menu “OPTIONS”

Within the menu “OPTIONS” (upper line in Netter), you may be able to configure colours and other representation characteristics of your schematisation. Within “2D grid options” you may modify the following items:

- 2D grid options / general: grid visible or not, colours;
- 2D grid options / Z-data / properties (see Figure 5.8: you can define yourself the scale for the z-terrain level data, the colour tone, the order, and to make the data visible or not);
- 2D grid options / dimensions (grid characteristics, not editable).

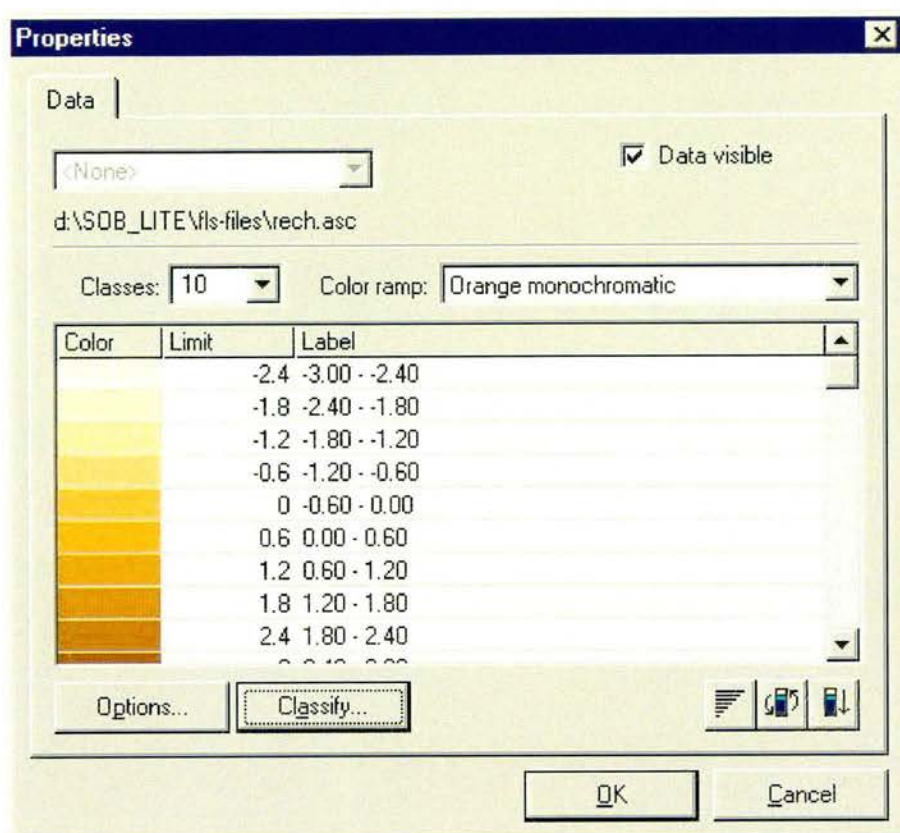


Figure 5.8 D2 grid options / Z-data / properties

If you want to delete a 2D grid, first make sure that the 2D grid node type is visible (Options / Network options / Network settings / Node settings / 2D grid visible: see Figure 5.9). Then you may delete it within the menu Edit Network / delete node.

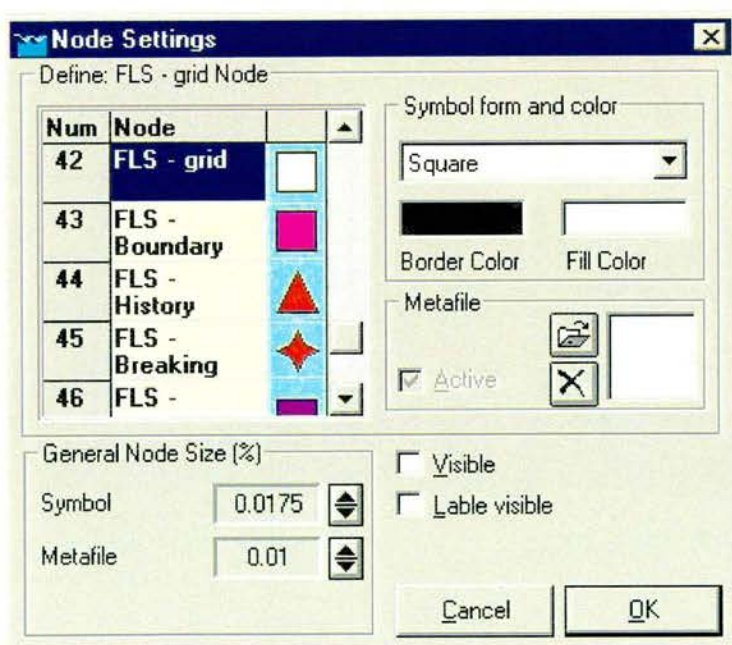


Figure 5.9 Options / Network options / Network settings / Node settings / 2D grid

Our example:

Define a 2D grid with the following characteristics:

```

nCols          100
nRows          20
xllCorner      257150
yllCorner      584000
CellSize       100
nodata_value   -999.000
Depths:
first 2 rows:  all 100 columns depth= -3
next 15 rows:  first 5 columns depth= -3, columns 6 to 84 depth=+3,
               columns 86 to 100 depth= -3
last 3 rows:   all 100 columns depth= -3

```

(See example of the associated import file reach.asc at appendix, at the end of this tutorial).

This grid (while Z-data is visible) is shown in Figure 5.10.

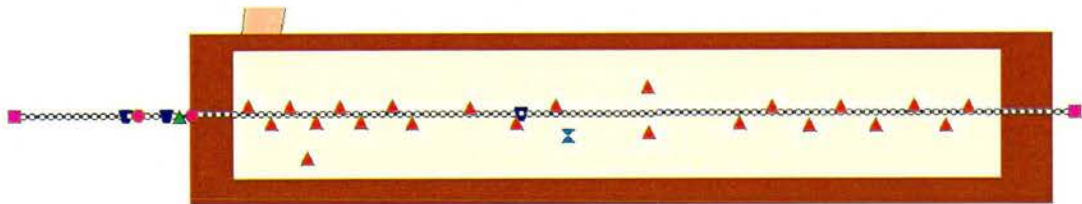


Figure 5.10 Grid and terrain levels combined with the 1D network used in this tutorial

Important notes related to the coupling 1D-2D:

Note 1:

When defining a 2D grid and a 1D grid, the system will automatically connect both grids at the connection and calculations points of the 1D schematisation that fall into the 2D grid area. If we define a few connection / calculations nodes on the 1D grid, while using a fine 2D grid, the coupling may be “poor”, that means the systems will interchange water only at those few coupling points and this may cause a course representation of the physically expected flooding pattern. In this case, we advise to use as much calculations points as necessary in order to give a good representation of the physical phenomena (one 1D connection or calculation point on each 2D cell). But check that there are not more than one calculation nodes per grid cell.

Note 2:

At the moment there are two options that can be used by the system to couple the channels with the neighbouring 2D area:

- flooding of the 2D cells starts when the water at the channel reaches the ground level of the cell (if the channel has dikes or levees rising above the ground level, they are taken as “permeable”).

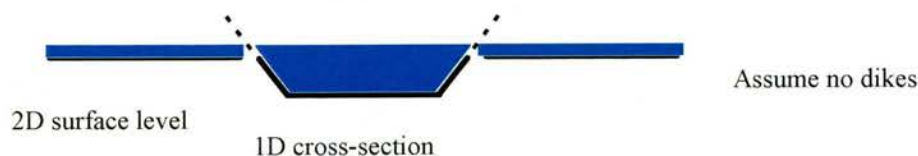


Figure 5.11 Permeable dikes

- flooding of the 2D cells starts when the highest level defined in the 1D cross-section is overtopped (see Figure 5.12). This is very suitable, in case the channel bed or walls lay higher than the ground level. This option requires special attention of the user when defining the 1D cross-sections.



Figure 5.12 Active dikes

5.3.2 2D - Boundary

(not specifically used in this tutorial)

Define the location of a 2D - Boundary while working in the Edit Network mode (see pink square, Figure 5.4). Give the boundary conditions in Model Data, by selecting the 1D of your 2D - Boundary. A discharge (m^3/s) or a water level, constant or variable in time can be specified at a 2D - Boundary. A 2D - boundary can be defined only at cells where no terrain level value has been defined (cells with -999.000 no-data value). When choosing a boundary variable in time you must give the data in table form including: Date (dd-mm-yyyy), Time (hh:mm:ss), Discharge (m) or water level (m above or below reference level) (general SOBEK- Time Series format).

5.3.3 2D - Boundary corner / 2D - Line Boundary

(not specifically used in this tutorial)

Define the location of two boundary corner nodes while working in the Edit Network mode (see blue rectangle in Figure 5.4). When these two corners are connected, the system understands that you want to give the same boundary condition to all the cells lying along the connection line (including the corners). These boundary corners can be defined only at cells where no terrain level value has been defined (cells with -999.000 no-data value). This also applies to the cells lying under the connection line.

In Network / Edit, select Branch: select 2D-Line Boundary, and then proceed to connect existing nodes (in this case the two Boundary corner nodes). The connection line must be horizontal or vertical; a diagonal boundary is not possible.

The values of the boundary parameter, discharge or water level, constant or time dependent are given in "Model Data"/2D - line boundary in the same way as in the normal 2D-Boundary. The given value will be assigned at each cell falling under the connection line, including the corners (For example: when defining a line boundary over three cells, if we specify a discharge of 60 m³/s to this line, the total discharge along the line will be 180 m³/s).

5.3.4 2D - Breaking dam

(not specifically used in this tutorial)

In this new system, breaking dams can be simulated in three ways. The first method is based on the use of the standard weir structure available in SOBEK-channel flow. This structure can be controlled by different types of controllers (see SOBEK-CF manual). The second method uses the 1D dam break branch. The third possibility is to use a 2D-Breaking dam formulation per 2D cell (As used by the existing Delft-FLS software). This last method is explained below.

Define the location of a 2D-Breaking Dam while working in the Edit Network mode (see red star in Figure 5.4). In "Model Data" you will be able to define the characteristics of the dam break.

Figure 5.13 shows the data to be provided when a linearly varying dam break occurs. In this case 30 minutes after the start of the computation, the terrain level of the cell starts to decrease linearly to stop 4 hours after the start of the computation at 2.50 m under the initial terrain level.

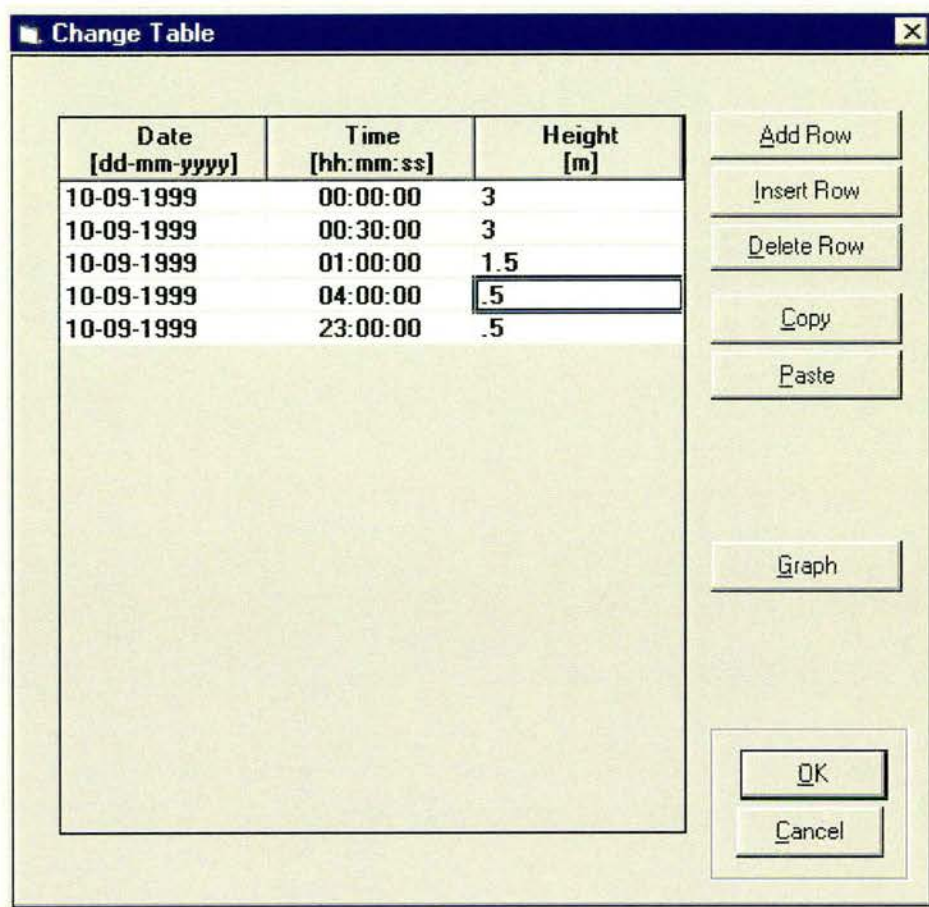


Figure 5.13 2D-Dam break. Non-linear breach-depth grow

5.3.5 2D - Initial Water Level Point

Define the location of a 2D-Initial Water Level Point while working in the Edit Network mode (see blue double-triangle in Figure 5.4). In “Model Data”, three sheets are available: the first one, non editable, shows the location of this point, the second sheet allows you to define the initial water level; the third one contains the default definitions.

It is very important to be careful when defining a 2D - Initial Water Level Point. The system allocates this level to all neighbouring (connected) cells that have a terrain level equal or lower than the given initial water level. Therefore, when you want to start with a certain area “dry”, choose the lowest cell in order to define the initial water level.

5.3.6 2D - History Node

Define the location of a 2D-History Node while working in the Edit Network mode (see red triangle in Figure 5.4). The location is defined while placing it on a cell. No input parameters are required. The frequency of output has already been defined at the Channel Flow / Output Settings (Settings Box in CMT).

In general all the 2D (requested) results will be stored in incremental files. Only at those locations where a 2D - History Node has been defined, the whole time series will be saved following the SOBEK-his format: this means the computed value of the variable at each (requested output) time step.

5.4 Simulations

After editing / finalising the schematisation and checking it for consistency, please leave the Schematisation box in CMT and save the case.

To proceed to start a simulation, click on the Simulation box in CMT. If errors are encountered, they will be reported immediately on screen and/or on file. These messages can be accessed through the CMT / Simulation Messages menu.

Once the simulation is completed you can proceed to view the results in maps and in charts.

5.5 Results

5.5.1 Results in Maps

The visualisation of 1D results has already been explained in the SOBEK-CF manual. Simultaneously with the visualisation of 1D results, the user can see the 2D results. In order to do so, enter the "Results in Maps" CMT-box. The Netter application will appear. Select File / Open Data / Overland Flow module "depths" or "results history stations".

The two-dimensional flooding "manual"-animation of the water depths varying in time, is based on results stored in incremental files (classes).

When one or several 1D-elements are selected, it is possible to observe time history graphs at those locations for the computed parameters (see Figure 5.14). The same could be done for the 2D- history stations.

Figure 5.15 shows a 2D map water depth output at a certain time step and a graph showing the water depth variation at selected 2D cells. This last graph is based on incremental information and it shows the moments where the water depth in one cell changes from one class to the next one while flooding proceeds (or to the previous one in case of drying processes).

5.5.2 Results in Charts

No special comments. See general instructions in SOBEK-CF / ODS-View manuals.

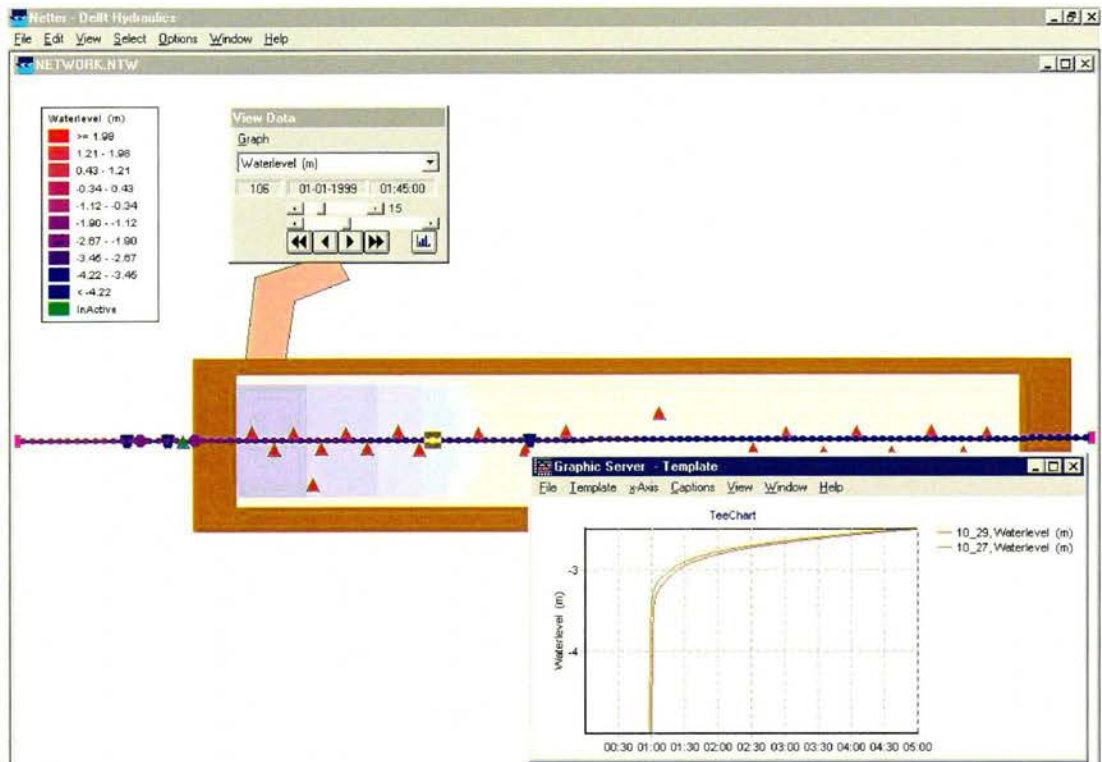


Figure 5.14 Results visualisation on maps; 1D + 2D map output for a certain time step and time series graph showing the water level history at two 1D locations

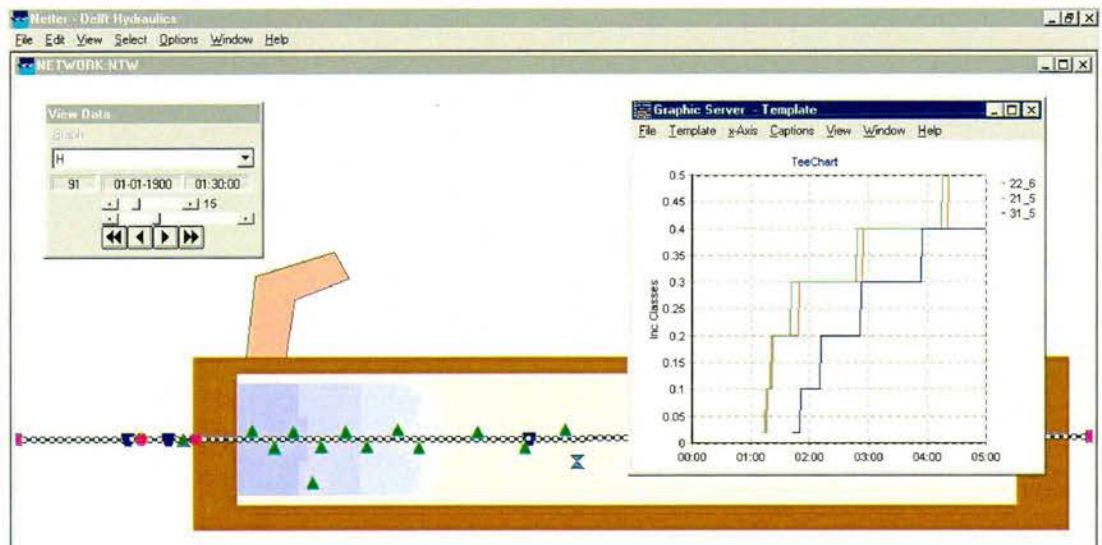


Figure 5.15 Results visualisation on maps; 2D map output for a certain time step and graph showing the water depth variation at selected 2D cells based on incremental information (this graph shows the moments where the water depth class-change)

APPENDICES

A Example grid file (test1.asc)

```

nCols      20
nRows      5
xllCorner  177367.5625
yllCorner  478413.96875
CellSize   100
nodata_value -999.000
-999 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 -999 -999
1 1 2 1 1 1 1 1 1 1.1 1.3 1.05 1 1 1 1 1 1 -999 -999
0 1 2 2 2 1 1 1 1 1.2 1.21 1 1 1 1 1 1 0 0
0 1 2 3 2 1 1 1 1 1 1 1 1 1 1 1 1 1 -999 -999
0 1 2 3 2 2 2 2 1 1 1 1 1 1 1 1 1 1 -999 -999

```

The heading of an asc-file contains the following information about the grid:

- the number of columns
- the number of rows
- x-coordinate of the bottom left corner of the grid
- y-coordinate of the bottom left corner of the grid
- cellsize of the grid elements (same for x and y size)
- the no_data value ('missing value'), usually -999 or -9999

For every cell the terrain level **BELOW** a certain reference level (depths).

B Background information on ID Dam Break reach

The following information is used to calculate the time-Area table (when generated automatically):

Info in initialization file or hard coded in program:

- max. allowed breach width for both types of dam (B_{\max} ; read from ini-file (\sob_lite\programs\flow\SBKEDIT.FNM)¹ under key [Switches] as entries “DamBreak1DMaxWidthSand” and “DamBreak1DMaxWidthClay”; defaults: 200 m for sand and 75 m for clay if no values are specified in the initialization file)
- max. number of steps for the t-A curve when the dam break grows in width ($n_{\max} = 20$)
- min. step size for generated t-A table ($s_{\min} = 5$ sec.)
- min. increase in area per time step for generated t-A table ($dA_{\min} = 0.01$ m²)
- percentage areal increase for 1st step on t-A curve to smooth linear part with logarithmic curve ($p = 10\%$)

Info supplied by the user through the UI:

- initial breach width (B_{ini})
- maximum breach depth (D_{\max})
- start of dam break (t_0)
- time to reach maximum breach depth (dt_1)
- type of dam (sand or clay)
- either maximum breach width (B_{\max}) or time to reach max. breach width (dt_{\max})

The type of dam determines the equations to be used (according to memo F.C.M. van der Knaap d.d. 23-05-2000 “Bresgroei Q2655”):

- for **sand**: $B(t) = 67 \cdot \log(t/522)$ and $t(B) = 522 \cdot 10^{B/67}$
- for **clay**: $B(t) = 20 \cdot \log(t/288)$ and $t(B) = 288 \cdot 10^{B/20}$

The info supplied by the user is checked on the following conditions:

- $B_{\text{ini}} > 0$
- $D_{\max} > 0$
- $B_{\text{ini}} + B(s_{\min} \cdot n_{\max}) \leq B_{\max} \leq \underline{B}_{\max}$
- $dt_1 + s_{\min} \cdot n_{\max} \leq dt_{\max} < t(\underline{B}_{\max})$

If all conditions are met, two initial time steps (the linear part) are calculated:

1st time step: $A_0 = 0$ m²

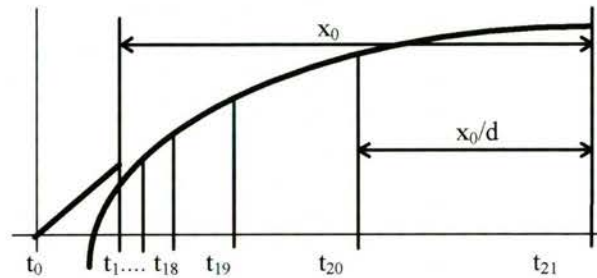
2nd time step: $t_1 = t_0 + dt_1$ and $A_1 = D_{\max} \cdot B_{\text{ini}}$

Then, based on the end condition, the last time step is calculated according to the applicable equation. The starting point for the equation ($t = 0$) is supposed to coincide with t_1 . NB: The total number of time steps is n_{\max} time steps + 2 initial time steps = 22.

22nd time step: $t_{21} = t_0 + dt_{\max}$ or $t_{21} = t_1 + t(B_{\max})$ and $A_{21} = D_{\max} \cdot B(t_{21} - t_1)$

¹ In the near future the initialization files used to store these values will be changed to SBKEDIT.INI (acc. to ARS 6175).

Now all intermediate time steps (3rd through 21st) can be calculated. Because of the equations being logarithmic functions, these time steps are not linearly divided over the remaining interval, but as shown in the figure below, thus producing more values for B in the region where the curve is steep.



When $x_0 = t_{21} - t_1$, $x_1 = t_{20} - t_1$ and d is a divider such that $d = x_0/(t_{21} - t_{20})$ then:

$$x_n = x_0 \cdot (1 - 1/d)^{n_{\max} - 1} \quad \text{and} \quad x_n \geq s_{\min}$$

Or:

$$d \geq 1 / (1 - (s_{\min}/x_0)^{1/(n_{\max} - 1)})$$

Based on this divider d the time steps t_{20} through t_2 can be calculated:

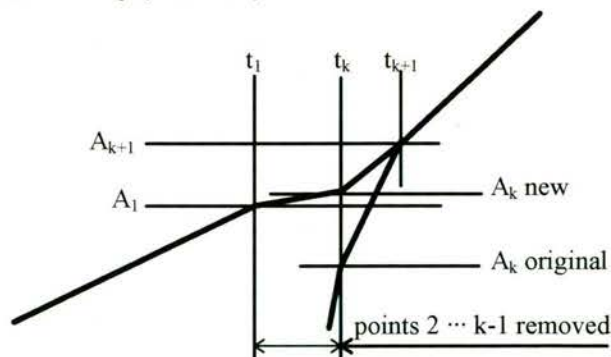
$$t_i = t_{i+1} + (t_1 - t_{i+1})/d$$

$$A_i = D_{\max} \cdot B(t_i - t_1)$$

Since the logarithmic functions will evaluate negative values when closing in to $t = 0$, there will often be some values (from t_2 upto t_k) calculated for the breach width smaller than B_{ini} . Of course the breach width can not decrease, so these values are removed from the table, except for the last one (t_k with A_k such that $A_{k+1} > A_1$), which is used to smoothen the linear part with the logarithmic curve.

A percentage p is introduced to determine the breach width value for this point according to the following rule:

$$A_k = A_1 + p \cdot (A_{k+1} - A_1)$$







WL | delft hydraulics

Rotterdamseweg 185
postbus 177
2600 MH Delft
telefoon 015 285 85 85
telefax 015 285 85 82
e-mail info@wldelft.nl
internet www.wldelft.nl

Rotterdamseweg 185
p.o. box 177
2600 MH Delft
The Netherlands
telephone +31 15 285 85 85
telefax +31 15 285 85 82
e-mail info@wldelft.nl
internet www.wldelft.nl

